



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA PODNIKATELSKÁ
ÚSTAV INFORMATIKY

FACULTY OF BUSINESS AND MANAGEMENT
INSTITUTE OF INFORMATICS

NÁVRH DÁTOVEJ ŠTRUKTÚRY PRE INFORMAČNÝ SYSTÉM CESTOVNEJ KANCELÁRIE

DATA STRUCTURE PROPOSAL FOR TRAVEL AGENCY'S INFORMATION SYSTEM

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

RÓBERT SOLÁRIK

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. JIŘÍ KŘÍŽ, Ph.D.

BRNO 2009

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Solárik Róbert

Manažerská informatika (6209R021)

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách, Studijním a zkušebním řádem VUT v Brně a Směrnicí děkana pro realizaci bakalářských a magisterských studijních programů zadává bakalářskou práci s názvem:

Návrh dátovej štruktúry pre informačný systém cestovnej kancelárie

v anglickém jazyce:

Data Structure Proposal for Travel Agency's Information System

Pokyny pro vypracování:

Úvod

Vymezení problému a cíle práce

Analýza problému a současné situace

Teoretická východiska práce

Vlastní návrhy řešení, přínos návrhů řešení

Závěr

Seznam použité literatury

Přílohy

Seznam odborné literatury:

- ECKER, C. Profesionální webdesign : techniky a vzorová řešení. 1. vyd. Brno: CP Books, 2005. 421 s. ISBN: 80-251-0547-4.
- GROFF, James R. a WEINBERG, Paul N. SQL : Kompletní průvodce. 2005. 936 s. ISBN 80-251-0369-2.
- HERNANDEZ, Michael J. Návrh databází, 2006. 408s. ISBN 80-247-0900-7.
- KOSEK, J. HTML. Tvorba dokonalých WWW stránek. Grada, 1998. ISBN 80-7169-608-0
- KŘÍŽ, J., DOSTÁL, P. Databázové systémy : studijní text 1. vyd. Brno : Akademické nakladatelství CERM, s.r.o., 2005. ISBN 80-214-3064-8.
- PALOVSKÁ, Helena. Databáze jako informační zdroj pro uživatele. 2004. 87s. ISBN 80-245-0720-X.

Vedoucí bakalářské práce: Ing. Jiří Kříž, Ph.D.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2008/2009.

L.S.

Ing. Jiří Kříž, Ph.D.
Ředitel ústavu

doc. RNDr. Anna Putnová, Ph.D., MBA
Děkan fakulty

V Brně, dne 25.05.2009

Abstrakt

Táto bakalárska práca sa zaoberá návrhom dátovej štruktúry pre informačný systém a webovú prezentáciu cestovnej kancelárie. Prvá časť práce analyzuje súčasnú situáciu a identifikuje možné problémy pri fungovaní spoločnosti. Na základe tejto analýzy je vytvorený vlastný návrh dátovej štruktúry pre informačný systém cestovnej kancelárie.

Kľúčové slová

Návrh databáze, dátová štruktúra, internetová aplikácia, informačný systém

Abstract

This bachelor's thesis deals with data structure development for travel agency's information system. First part of this thesis analyses current situation and possible problems within the company. The data structure proposal for the travel agency's information system is based on this analysis.

Keywords

Database design, data structure, internet application, information system

Bibliografická citácia práce

SOLÁRIK, R. Návrh dátovej štruktúry pre informačný systém cestovnej kancelárie. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2009. 51 s.
Vedoucí bakalářské práce Ing. Jiří Kříž, Ph.D.

Čestné prehlásenie

Prehlasujem, že predložená bakalárska práca je vypracovaná samostatne na základe uvedenej literatúry a pod vedením vedúceho práce. Prehlasujem, že citácia použitých zdrojov je úplná a že som v práci neporušil autorské práva (v zmysle zákona č. 121/2000 Zb. o práve autorskom a o právach súvisiacich s právom autorským.)

V Brne dňa 20. 5. 2009

.....

Róbert Solárik

Pod'akovanie

Chcel by som poďakovať vedúcemu práce, pánovi Ing. Jiřímu Křížovi, Ph.D. za čas, ktorý mi venoval a cenné rady, ktoré mi poskytol.

Obsah

Obsah	7
1 Úvod.....	9
2 Vymedzenie problému a ciele práce	10
2.1 Vymedzenie problému	10
2.2 Ciele práce.....	10
3 Analýza problému a súčasnej situácie	11
3.1 Profil spoločnosti Solartour.....	11
3.1.1 O spoločnosti	11
3.1.2 História a súčasnosť	11
3.1.3 Kontaktné údaje	12
3.2 Popis súčasnej situácie	12
3.2.1 Objednávky na diaľku.....	13
3.2.2 Webová stránka.....	13
3.2.3 Elektronická komunikácia so zákazníkmi	14
3.2.4 Komunikácia s províznymi predajcami	14
3.2.5 Optimalizácia pre vyhľadávače	15
3.2.6 Evidencia objednávok.....	15
3.3 Analýza problému	15
3.3.1 Zákazník.....	16
3.3.2 Administrátor / pracovník cestovnej kancelárie	18
4 Teoretické východiská práce	20
4.1 Informácie a Informačné systémy	20
4.1.1 Dáta a informácie.....	20
4.2 Typy databáz	21
4.2.1 Operačné databáze	21
4.2.2 Analytické databáze	21
4.3 Databázové modely	22
4.3.1 Hierarchický model.....	22
4.3.2 Sieťový model.....	22
4.3.3 Relačný model	23
4.3.4 Ostatné modely	25
4.4 Proces tvorenia databázy	25

4.4.1	Tradičné metódy logického návrhu	26
4.4.2	Normálne formy	27
4.5	Technológie	28
4.5.1	MySQL	28
4.5.2	IIS	28
4.5.3	ASP.NET	28
4.6	Diagramy	29
4.6.1	ER diagram	29
4.6.2	EPC diagram	29
4.6.3	Vývojový diagram - Flowchart	29
5	Vlastné návrhy riešenia a ich prínosy	30
5.1	Časový harmonogram	30
5.2	Logický návrh	32
5.2.1	Textové dáta	33
5.2.2	Nastavenia systému	34
5.2.3	Riadenie prístupu a správa užívateľov	34
5.2.4	Regióny	35
5.2.5	Modul správy zájazdov	36
5.2.6	Modul správy objednávok	37
5.2.7	Zľavy	38
5.3	Prínosy pre cestovnú kanceláriu	39
6	Záver	40
7	Zoznam použitej literatúry	41
7.1	Monografické zdroje	41
7.2	Elektronické zdroje	41
8	Zoznam obrázkov a tabuliek	43
9	Prílohy	44
9.1	Webová stránka postavená na navrhovanej dátovej štruktúre	44
9.2	Informačný systém postavený na navrhovanej dátovej štruktúre	44
9.3	Fyzická implementácia	45
9.4	Zdrojový kód na vytvorenie dátovej štruktúry	46

1 Úvod

Tak ako v iných odvetviach, aj v odvetví cestovného ruchu je konkurencia cestových kancelárií vysoká. Všetky subjekty sa snažia minimálne udržať si svoju pozíciu z minulého roka, respektíve spoju pozíciu na trhu zlepšiť.

To spôsobuje, že sa konkurenčný boj presúva napríklad aj na pole internetu. V dnešnej dobe už nestačí mať webovú stránku na ktorej bude uvedený kontakt, ale firmy investujú do moderných a interaktívnych prezentácií, ktoré sú určitou formou elektronických obchodov. Tento spôsob predaja potom umožňuje firme osloviť nových zákazníkov a dostať sa na nové trhy.

Vo svojej bakalárskej práci sa zaoberám návrhom dátovej štruktúry na základe ktorej bude možné vybudovať modernú webovú prezentáciu a informačný systém pre cestovnú kanceláriu.

Prvá časť práce je venovaná analýze problému a súčasnej situácie. V tejto časti analyzujem stav v akom sa firma momentálne nachádza a s akými problémami sa pri svojom chode stretáva. Výstupom z analýzy je konštatovanie, že situáciu je možné zlepšiť a je možné vytvoriť dátovú štruktúru na základe ktorej bude vytvorená nová webová stránka spoločnosti a informačný systém.

Nasleduje časť zameraná na teoretické východiská potrebné k vyriešeniu situácie. Nejedná sa o úplné a vyčerpávajúce informácie, ale o stručný prehľad teoretických základov, ktoré som pri vlastnom návrhu potreboval.

Posledná časť práce predstavuje návrh riešenia problému. Na základe analýzy a pomocou nadobudnutých teoretických znalostí som navrhol dátovú štruktúru na základe ktorej bude vytvorená moderná webová stránka cestovnej kancelárie a informačný systém na správu zájazdov a objednávok. Dá sa povedať, že sa jedná o akýsi elektronický obchod so zájazdmi. Na koniec sú uvedené prínosy pre cestovnú kanceláriu a stručné zhodnotenie doterajších výsledkov.

2 Vymedzenie problému a ciele práce

2.1 Vymedzenie problému

Cestovná kancelária sa pri svojej činnosti čoraz viac stretávala s požiadavkou o on-line rezervácie, respektíve objednávku dovolenkových pobytov od svojich zákazníkov a provízných predajcov. Z toho vyplynula požiadavka vytvoriť systém, ktorý by bol schopný evidovať obsadenosť, respektíve dostupnosť jednotlivých ubytovacích kapacít a ich aktuálnej ceny v jednotlivých termínoch. Tieto informácie by mali byť pokiaľ možno vždy aktuálne a dostupné on-line na internetových stránkach spoločnosti s možnosťou kalkulácie celkovej ceny za zájazd a s možnosťou on-line objednávky respektíve tlačou cenovej ponuky. Druhotným problémom je potom správa týchto on-line objednávok a proces ich vybavovania a evidencie.

2.2 Ciele práce

Prvoradým cieľom mojej bakalárskej práce je navrhnuť dátovú štruktúru na základe ktorej bude možné vytvoriť on-line informačný systém na správu objednávok, zájazdov a služieb v jednotlivých krajinách, letoviskách a ubytovacích kapacitách. Ďalej bude z tejto dátovej štruktúry vychádzať webová stránka cestovnej kancelárie s možnosťou on-line objednávky, respektíve rezervácie a s možnosťou cenovej kalkulácie kompletnej ceny zájazdu vrátane povinných či nepovinných príplatkov a vrátane odpočítania aktuálnych zliav. Táto bakalárska práca má za cieľ zjednodušiť a zefektívniť prácu cestovnej kancelárie a sprístupniť aktuálne informácie potenciálnym klientom alebo províznym predajcom.

3 Analýza problému a súčasnej situácie

3.1 Profil spoločnosti Solartour

3.1.1 O spoločnosti

Cestovná kancelária SOLARTOUR sa už od svojho vzniku v roku 1990 špecializuje hlavne na letné pobytové zájazdy a poznávacie zájazdy.

3.1.2 História a súčasnosť

V prvých rokoch svojho fungovania sa zameriavala na organizovanie autobusových zájazdov do Talianska a na organizovanie krátkych poznávacích výletov napríklad do Rakúska.

Neskôr svoju činnosť rozšírila a začala organizovať autobusové zájazdy do Španielska a letecké zájazdy na ukrajinský Krym. Pri rozširovaní svojho poľa pôsobnosti v polovici deväťdesiatych rokov začína cestovná kancelária organizovať letné dovolenkové pobyty okrem Španielska, Talianska a Ukrajiny do Grécka, Chorvátska a Bulharska.

Cestovná kancelária Solartour momentálne organizuje letné pobytové zájazdy do Bulharska a Chorvátska, ale dlhé už roky sa špecializuje na organizovanie autobusových zájazdov do Grécka. Hlavne na Olympskú riviéru, polostrov Chalkidiki a ostrovy Thassos a Korfu.

Sídlom spoločnosti je mesto Martin (Slovenská republika). V meste sa pravdepodobne momentálne nenachádza žiadna iná cestovná kancelária, ale samozrejme tu existuje množstvo cestových agentúr, ktoré provízne predávajú produkty iných cestovných kancelárii. Solartour je z právneho hľadiska reprezentované fyzickou osobou – živnostníkom.

Cestovná kancelária počas svojej existencie vystriedala niekoľko kancelárskych priestorov a momentálne sa jej kancelária nachádza v obchodnom dome Prior v centre mesta v blízkosti pešej zóny. Mimo činnosti tour operátora – organizovania zájazdov – je možné v kancelárii kúpiť medzinárodné autobusové lístky a letenky, lístky na kultúrne podujatia prostredníctvom systému Ticketportal alebo zájazdy konkurenčných cestovných kancelárii, a to hlavne zájazdy do krajín do ktorých CK Solartour sama zájazdy neorganizuje.

3.1.3 Kontaktné údaje

Poštová adresa:

SOLARTOUR, cestovná kancelária

Námestie SNP 2 (OD Prior)

036 01 Martin

Slovenská republika

Elektronické adresy:

solartour@solartour.sk

www.solartour.sk

Telefonické spojenie:

telefón:

+421 43 4223 820

+421 43 4223 839

fax:

+421 43 4135 663

3.2 Popis súčasnej situácie

Základným propagačným materiálom cestovnej kancelárie sú tlačené katalógy v ktorých klienti nájdu popisy jednotlivých letovísk, ďalej popisy ubytovacích kapacít a cenníky pre jednotlivé termíny. Objednávky môžu klienti vykonávať buď osobne, prostredníctvom províznych predajcov alebo na diaľku - telefonicky, respektíve mailom. Pre osobný predaj je tento systém bezproblémový, klienti môžu osobne prísť do kancelárie, orientačne sa informovať o obsadenosti, prezrieť si katalóg, vybrať si z ponuky pobytov a zájazd uhradiť.

Na základe tlačeného katalógu bola teda vytvorená webová stránka spoločnosti, ktorá kopírovala samotný katalóg. To znamená, že internetová stránka je v podstate statická, aj keď je založená na redakčnom systéme v ktorom je možné editovať jednotlivé články. Podobne ako v tlačenom katalógu je cenník v podobe tabuľky bez logického uloženia dát v určitej štruktúrovanej forme ako napríklad databáza. A teda všetky úpravy na webovej stránke sú možné len v textovej forme cez administráciu s editorom článkov typu „WYSIWYG“.

3.2.1 Objednávky na diaľku

Pri riešení objednávok „na diaľku“ momentálne cestovná kancelária nepoužíva žiadny informačný systém a väčšina informácií a dát s ktorými pracuje a ktoré je potrebné uchovávať sú nejakým spôsobom katalogizované v papierovej forme, respektíve v tabuľkách programu Excel. Počas trvania sezóny tak opakovane vznikala situácia, že klienti telefonovali alebo písali a informovali sa o voľných kapacitách. Tento systém spôsoboval značné personálne nároky a preto sa ho rozhodla cestovná kancelária zmeniť.

Pri každom klientovi pri jeho požiadavke bolo potrebné kontaktovať klienta a prediskutovať voľné termíny a ceny, vytvoriť cenovú kalkuláciu a ručne vyrobiť zmluvu o obstaraní zájazdu (prihlášku). Tieto procesy boli veľmi časovo náročné a kvôli prvej neinformovanosti potenciálnych zákazníkov bol pomer zákazníkov, ktorí sa len informovali k tým čo naozaj objednávku uskutočnili veľmi nízky.

3.2.2 Webová stránka

Tým, že webová prezentácia je podstatne len určitým spôsobom formátovaný text, dáta nie sú uložené žiadnym iným štruktúrovaným spôsobom. Teda neexistuje štruktúra krajín, letovísk a ubytovacích zariadení. Cenníky sú tiež len formátovanými tabuľkami s legendou a teda nemôžu byť použité pri žiadnom ďalšom počítaní.

Nevýhody takehoto systému sú teda zrejmé. V prvom rade je to časovo náročná a zložitá aktualizácia. V prípade zmeny je síce možné napríklad zmeniť formátovanie bunky tabuľky a tým dať najavo, že kapacita je vypredaná, alebo zmenením ceny dať najavo ponuku „last minute“, ale ďalej nie je možné už tieto dáta nijak využívať. Nie je možná ani plošná zmena cien. Takáto situácia nastáva na začiatku sezóny, kedy je napríklad na ubytovanie poskytovaná pri všetkých kapacitách a termínoch zľava za skorý nákup. Táto zľava sa však mení s blížiacim sa termínom odchodu. Meniť všetky tabuľkové cenníky niekoľkokrát v priebehu sezóny je teda pre cestovnú kanceláriu skoro nemožné a percentuálne počítanie zľavy klientom je zase nepraktické.

Ďalšou nevýhodou je absencia cenovej „kalkulačky“. Celková cena zájazdu pozostáva z viacerých častí. Základu službou je ubytovanie, ďalej doprava, strava, poistenie a nakoniec rôzne iné príplatky (za domáce zvieratá, klimatizáciu, balkón, výhľad na more, ...) a zľavy. Z tabuľkového cenníka niekedy nie je zrejmé, ktoré príplatky sú platné ku ktorým termínom, ktoré služby je možné kombinovať a teda

klienti nie sú často krát schopní sami vypočítať celkovú cenu. Toto spôsobuje jednak odradenie potenciálnych zákazníkov a zároveň náklady spojené s vysvetľovaním ponuky a vytváraním možných alternatív spolu s klientom, či už po telefóne, alebo elektronickou poštou.

Kvôli spomínanej forme webovej stránky – len určitým spôsobom formátovaný text - nie je možné vytvoriť na stránke vyhľadávanie, respektíve filtrovanie zájazdov. Existujúce fulltextové vyhľadávanie síce umožňuje prehľadávať jednotlivé články, ale pre vyhľadanie zájazdu podľa určitých kritérií je viac menej nepoužiteľné. Interpretácia výsledkov takéhoto fulltextového vyhľadávania je tiež otázna, pretože na stránke sa v skutočnosti môže vyskytovať ponuka na zájazdy letecky do Grécka, ale nevhodne zvolenými kľúčovými slovami pri vyhľadávaní túto ponuku klient nemusí nájsť. Pri výsledkoch takisto nie je zrejmé či je to celá ponuka, alebo sa na stránke nachádzajú aj iné, podobné ponuky.

3.2.3 Elektronická komunikácia so zákazníkmi

V súčasnosti tiež nie je možné rozposielanie hromadných mailov, keďže v terajšom systéme nie je možnosť viesť zoznam účastníkov, ktorým by bola hromadná pošta odoslaná a nedá sa na odber takýchto informácií prihlásiť.

S hromadnými mailmi úzko súvisí aj problematika propagácia akciových, respektíve Last Minute ponúk. Keďže stránka nemá žiaden systém na evidenciu týchto ponúk, jedinou možnosťou je zmena textov v jednotlivých sekciách stránky. Ak aj pracovník cestovnej kancelárie zmení cenu v tabuľkovom cenníku a prípade zvýrazní túto cenu zmenou formátovania, stále sa táto ponuka nijak inak neprezentuje. Pracovníci kancelárie sa po minulé sezóny snažili udržiavať na úvodnej stránke aktuálne informácie o Last Minute ponukách, ale tie následne neboli podporené zmenou cenníkov pri jednotlivých kapacitách čo viedlo k neprehľadnosti a tým k častým telefonátom s prosbami o vysvetlenie konkrétnej ponuky.

3.2.4 Komunikácia s províznymi predajcami

V poslednom období tiež narastá záujem od províznych predajcov zaradiť ponuku cestovnej kancelárie na ich vlastné stránky. Aby bolo možné poskytnúť cestovným agentúram tieto dáta, je nutné najskôr takto štruktúrované dáta mať k dispozícii. V súčasnosti teda neponúka produkty cestovnej kancelárie Solartour žiadny on-line predajca zájazdov na internete, pretože spoločnosť momentálne nevie poskytnúť

aktuálne dáta o voľných kapacitách a aktuálnych cenách v štruktúrovanej forme (webové služby, XML, ...).

3.2.5 Optimalizácia pre vyhľadávače

Optimalizácia pre webové vyhľadávače je na slabej úrovni. Webová stránka obsahuje chyby v zdrojovom kóde, nespĺňa štandardy organizácie W3C a nemá správnu štruktúru dokumentu (nadpisy a odseky). URL jednotlivých stránok neobsahuje kľúčové slová, ktoré by mohli viesť k zvýšeniu pozície vo výsledkoch jednotlivých vyhľadávačov. Texty v jednotlivých článkoch tiež neobsahujú kľúčové slová, na ktoré by sa cestovná kancelária mohla zamerať.

3.2.6 Evidencia objednávok

Posledným, ale nie nepodstatným problémom pre cestovnú kanceláriu je spracovávanie a evidencia realizovaných objednávok „na diaľku“. Ak zákazník poslal svoju objednávku mailom, alebo ju nadiktoval pracovníkovi cestovnej kancelárie telefonicky, nemohol sledovať stav svojej objednávky a teda nevedel, v akom štádiu sa jeho objednávka nachádza. Musel sa spoliehať na dodržanie termínov cestovnou kanceláriou alebo sa mohol informovať telefonicky, alebo ďalším mailom. Toto však značne zdržovalo pracovníkov, pretože ich klient väčšinou vyrušil od inej činnosti. Pracovník najskôr musel zistiť informácie o objednávke a až potom mohol reagovať. Počas sezóny musela cestovná kancelária zamestnať brigádnika, ktorý odpovedal na podobné dotazy. Takýto spôsob zisťovania stavu objednávky, prijatie, respektíve neprijatie platieb a iné bol nepríjemný tak pre cestovnú kanceláriu ako aj pre samotných zákazníkov.

3.3 Analýza problému

Na základe súčasnej situácie a zistených skúseností a problémov z predchádzajúcich rokov môžeme problém rozdeliť do dvoch základných skupín. Výstupom tejto práce bude databázová štruktúra, ktorá bude základom pre dve aplikácie. Každá z nich bude riešiť základný problém z inej perspektívy.

V prvom rade pôjde o pohľad a potreby zo strany zákazníka a v rade druhom o potreby administrátora, respektíve pracovníkov cestovnej kancelárie. Požiadavky na dátovú štruktúru budú nasledovné:

3.3.1 Zákazník

Dátová štruktúra by mala byť navrhnutá tak, aby z bolo možné čerpať všetky potrebné údaje pre webovú stránku cestovnej kancelárie.

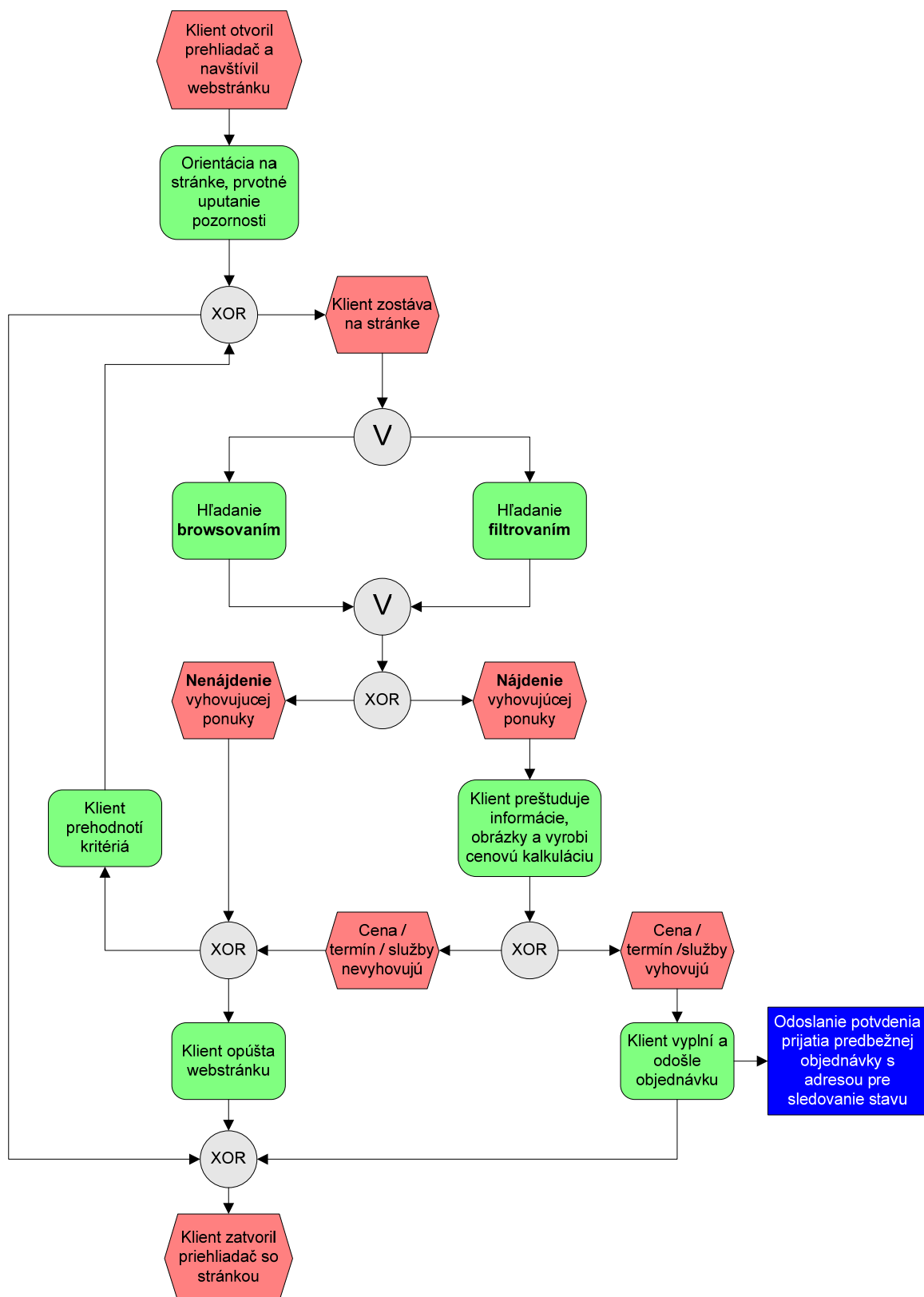
Na základe vytvorenej ponuky zájazdov (pracovníkmi CK) bude mať zákazník na stránke možnosť prezerat' jednotlivé „stránky elektronického katalógu“ podľa krajín a letovísk. Ďalej bude mať zákazník možnosť jednotlivé zájazdy filtrovať podľa krajiny, letoviska, ubytovacieho zariadenia, termínu, dopravy, možných spôsobov stravovania či typu ubytovania (štúdio pre rôzny počet izieb, apartmány, hotelové izby, ...)

K zákazníkom by sa mali dostať vždy aktuálne ceny a dostupné ubytovacie kapacity. Zákazník by mal teda mal byť schopný v reálnom čase jednoducho zistiť v ktorých termínoch je dostupný zájazd o ktorý má záujem. Mal by byť bez asistencie pracovníkov cestovnej kancelárie schopný zistiť aká je aktuálna cena celého pobytu – aká je základná katalógová cena, ale sú povinné či nepovinné príplatky, či aké zľavy sú momentálne platné.

Na základe týchto informácií by si zákazník mal vytvoriť cenovú kalkuláciu, ktorú bude mať možnosť odoslať ako nezáväznú objednávku. Po odoslaní objednávky a by mal byť zákazník schopný sledovať stav vybavovania svojej objednávky on-line s tým, že bude informovaný prostredníctvom elektronickej pošty o každej zmene. Konkrétne: Prijatie na spracovanie, Potvrdenie a rezervácie, Potvrdenie rezervácie, Prijatie platby, Zaslanie prihlášky, Prijatie podpísanej prihlášky, Odoslanie pokynov, respektíve voucheru.

Zákazník by tiež mal mať možnosť prihlásiť sa k odberu noviniek, aktuálnych ponúk či iných informácií (napríklad pozvánka na výstavu, ankety, súťaže, ...) na svoju emailovú schránku.

V nasledujúcom diagrame nie je zahrnutá zahrnutý proces kontroly stavu objednávky. Je to z dôvodu, že po odoslaní predbežnej objednávky dostane zákazník emailovú správu s adresou, kde môže sledovať stav svojej objednávky a teda cez verejnú prezentáciu sa ku kontrole stavu objednávky nie je možné dostať. V diagrame taktiež chýba proces prihlásenia sa k mailing listu, pretože návštevník tak bude môcť urobiť kedykoľvek a kdekoľvek na stránke sa práve bude nachádzať.



Obrázok 1 EPC diagram - očakávané správanie zákazníka na web stránke

3.3.2 Administrátor / pracovník cestovnej kancelárie

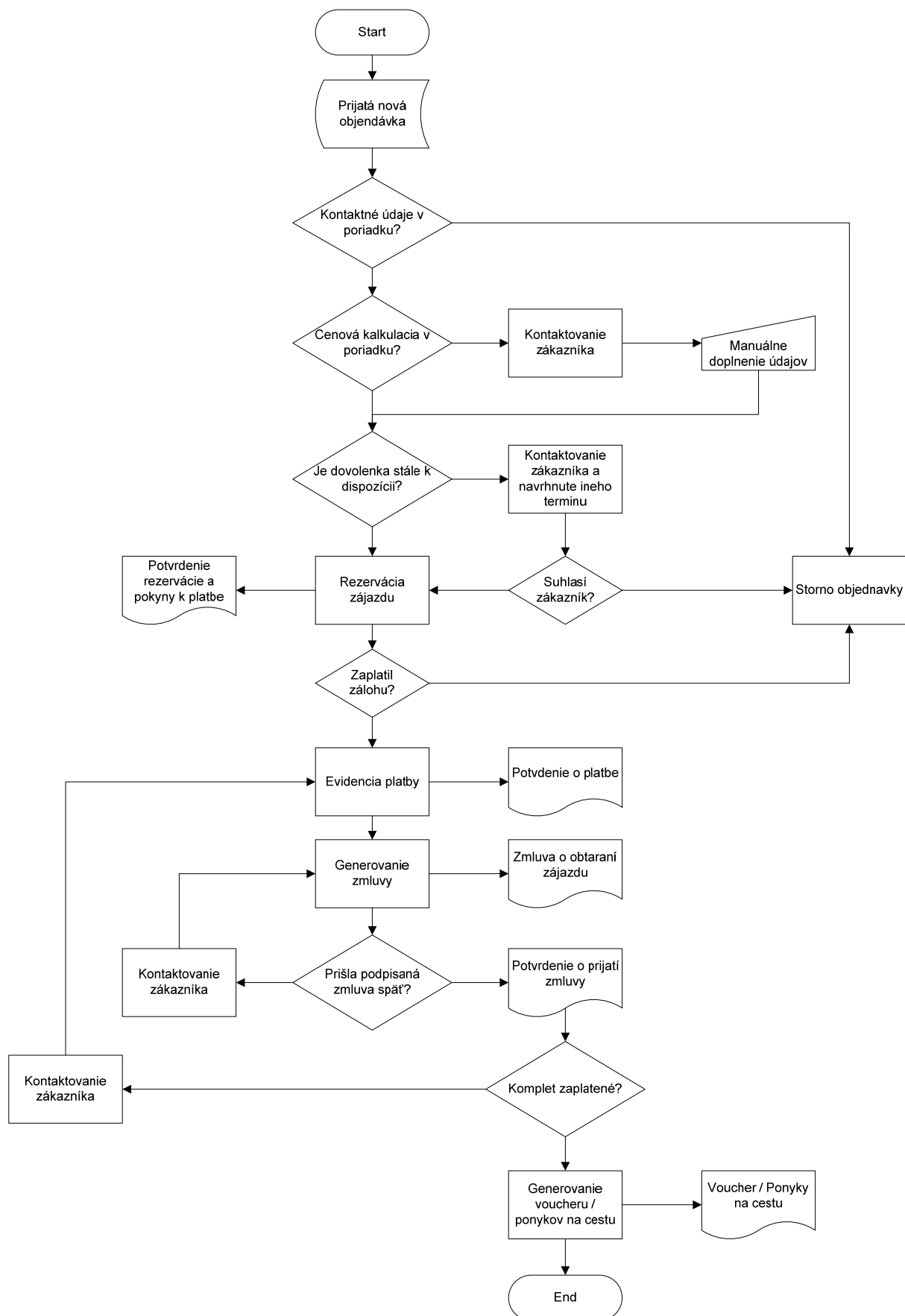
Na druhej strane sa nachádza informačný systém, ktorý bude spravovať texty a evidovať zájazdy a objednávky, ktoré boli vytvorené on-line na webovej stránke.

V prvom rade je potrebné, aby bola dátová štruktúra pripravená na uloženie informácií o konkrétnych zájazdoch. Jedná sa o hierarchickú štruktúru začínajúcu Krajinou, Letoviskom, pokračujúc Hotelom (ubytovacím zariadením) až po samotný zájazd. Zájazd by sa dal chápať ako „cenník“ určitého hotela s určitou dopravou pre určité termíny.

Pracovníci cestovnej kancelárie by mali byť schopný vytvárať, editovať aj mazať všetky časti tejto štruktúry. Dátová štruktúra musí počítat' aj s uložením iných textov ako tých priamo spojených so zájazdmi. Sú to napríklad všeobecné podmienky, kontakty, atd. Z toho vyplýva že dátová štruktúra musí umožňovať prostredníctvom informačného systému kompletné ovládanie obsahu webovej stránky.

Druhou a podstatou úlohou je správa prijatých on-line objednávok. Databáza musí samozrejme ukladať *Kto* objednávku odoslal, *Kedy* ju odoslal, o aké *Ubytovacie zariadenie* má zákazník záujem, v akom *Termíne*, aké *Služby* si objednáva a v akom *Množstve*. Ďalej by mal mať administrátor k dispozícii kontaktné údaje, informácie o spolucestujúcich vrátane dátumov narodenia. Na základe týchto informácií je cestovná kancelária schopná schváliť alebo upraviť cenovú kalkuláciu a zájazd potenciálnym klientom zarezervovať (či sa už jedná o vlastné zájazdy alebo o zájazdy partnerských cestovných kancelárií). V prípade zaplatenia zálohy klientom by malo byť možné z dát v informačnom systéme generovať zmluvu o obstaraní zájazdu, evidovať jednotlivé platby klienta, evidovať odoslanie a prijatie podpísanej prihlášky a odoslanie voucheru či pokynov na cestu.

Nasledujúci diagram zobrazuje proces vybavovania objednávky. Je dôležité pomenovať jednotlivé fázy tohto procesu a určiť ich prepojenia, aby bolo možné navrhnuť dátovú štruktúru, na základe ktorej bude možné vybudovanie informačného systému, ktorý by mal zjednodušovať prácu v cestovnej kancelárii. Diagram tiež ukazuje aké výstupy by malo byť možné generovať v konkrétnych fázach vybavovania objednávky.



Obrázok 2 Flowchart - Vybavovanie objednávky

4 Teoretické východiská práce

Bakalárska práca vychádza z teoretických poznatkov, ktoré som počas doterajšieho štúdia získal. Preto sa v tejto kapitole nachádza stručný prehľad teórie z oblastí, ktoré som pri svojej práci využíval.

4.1 Informácie a Informačné systémy

Hlavnou úlohou informačného systému je poskytovať svojim užívateľom potrebné informácie v požadovanom rozsahu, čase, detailoch a forme.

Za informačný systém sa považuje systém, ktorý určitým spôsobom zbiera a organizuje dostupné dáta, triedi ich, spracúva a na ich základe poskytuje užívateľom informácie. Informačný systém nemusí byť nutne v podobe aplikačného software, ale napríklad aj v papierovej forme. Zmyslom informačného systému je aj redukcia nepotrebným dát, prenos, archivácia alebo aj distribúcia dát.

4.1.1 Dáta a informácie

Slová „Dáta“ a „Informácie“ sa môžu na prvý pohľad zdať synonymami, ale v skutočnosti majú úplne odlišný význam.

Informácie sú dáta, ktoré sú určitým spôsobom spracované, sú pre užívateľa zrozumiteľné a použiteľné. Informácie sa stále menia. Sú dynamické v tom zmysle, že sa menia tak, ako sa menia dáta, ktoré sú fyzicky uložené v databáze. Takisto sa informácie menia podľa v zmysle, že z rovnakých dát sa rôznym spracovaním dajú získať rozdielne informácie. Dáta, ktoré môžu mať rozdielne podoby (text, číslo, obrázok, ...) bez správneho spracovania nedávajú žiadny význam a preto kým sa dáta zmenia na použiteľné a zmysluplné informácie, musia sa spracovať. (4)

Pre lepšie pochopenie uvádzam stručnú výpis charakteristík:

Dáta:

- Vyjadrenie určitej hodnoty / skutočnosti formálnym spôsobom
- Pevne dané vyjadrenie - text, číslo, obrázok, ... – ktoré je fyzicky zaznamenané
- Bez správneho spracovania neinterpretovateľné

Informácie:

- Dáta spracované do interpretovateľnej formy
- Zmysluplná poznatky o dátach vrátane vzťahov medzi nimi

Informačné systémy, respektíve databáze sú navrhované tak, aby poskytovali ich používateľom relevantné informácie. Toto je možné dosiahnuť iba vtedy, ak sa v databáze nachádzajú správne štruktúrované dáta. (4)

*„Dáta sú to čo **ukladáte**, informácie sú to, čo **získavate**.“*

(4, s. 67)

4.2 Typy databáz

Pri správe databáz môžeme hovoriť o dvoch základných typoch. Jedná a o operačné databáze a analytické databáze.

4.2.1 Operačné databáze

Operačné databáze sa primárne používajú v on-line spracovaní transakcií (**online transaction processing - OLTP**). Bežne tento typ databáz používajú tisíce firiem po celom svete v situáciách, kedy je potrebné zhromažďovať, meniť, respektíve spracovať dáta každý deň. Dáta uložené v databáze sa neustále menia, teda sú dynamické a vždy reprezentujú aktuálnu situáciu v podniku. Tento typ databáz využívajú firmy v premyslenom odvetví, maloobchody, nemocnice, klinky a podobne, pretože ich dáta sa neustále menia. (4)

4.2.2 Analytické databáze

Analytické databáze sa naproti tomu využívajú hlavne tam, kde je potrebné ukladať a vyhľadávať v starších a časovo závislých dátach. Sú primárne využívané na on-line analytické spracovanie (**online analytical processing - OLAP**). Tento typ databázy ukladá dáta ktoré sú statické – väčšinou historické a teda v priebehu času sa už nemenia. Analytická databáza je výhodná v prípade, že je to potrebné sledovať trendy, respektíve iné štatistické hodnoty za dlhšie časové obdobie a na ich základe sa dajú robiť strategické rozhodnutia alebo odhadovať budúci vývoj.

Analytické databázy často získavajú dáta hlavne z operačných databáz, takže medzi nimi môže byť určitý stupeň prepojenia. Operačné a analytické databázy však

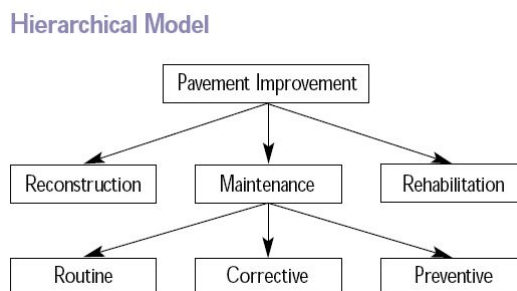
plnia úplne odlišné úlohy a preto si aj navrhovanie ich štruktúry vyžaduje radikálne odlišné postupy. (4)

4.3 Databázové modely

Databázové modely ako aj ostatné odvetvia informačných technológií prešli určitým vývojom. V súčasnosti je najpoužívanejším databázovým modelom relačný model. Napriek tomu tu uvediem aj stručný prehľad dvoch predchádzajúcich prístupov, aby sme mali aspoň základnú predstavu o tom čo viedlo k jeho vývoju.

4.3.1 Hierarchický model

V tomto type databáz sú dáta štruktúrované hierarchicky do podoby obráteného stromu. Vzťahy sú v tomto modeli reprezentované termínmi „rodič“ a „potomok“. V tomto type môže mať jeden potomok len jedného rodiča, ale rodič môže mať niekoľko potomkov.



Obrázok 3 Hierarchický databázový model

Hierarchické databáze sa používali hlavne v 70. rokoch a najlepšie sa hodili pri ukladaní na kazetovo páskových jednotách používaných v sálových počítačoch. Kvôli narastajúcim problémom s redundanciou dát a problémom s komplexnosťou vzťahov medzi dátami bolo jasné, že je potrebné vyvinúť nový databázový model. V súčasnosti sa však hierarchický model ukladania dát stále používa napríklad pri populárnom formáte XML. (4)

4.3.2 Sieťový model

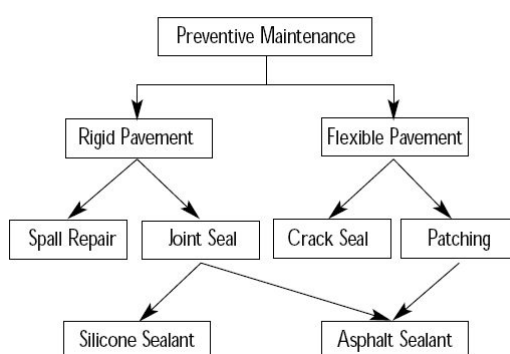
Sieťová databáza bola navrhnutá ako pokus k vyriešeniu problémov s hierarchickým modelom. Podľa Michaela J. Hernandez, je sieťový databázový model vyjadrený v pojmoch „uzlov“ a „množinových štruktúr“.

Uzol reprezentuje súbor určitých záznamov a **množinová štruktúra** reprezentuje a zariaduje vzťahy v sieťovej databáze. Množinová konštrukcia umožňuje

vzťahy 1:N takže jeden vlastník môže byť vo vzťahu s jedným ale aj viacerými záznamami v uzle člen.

Aj keď bol sieťový databázový model určite krokom vpred, stále existovali databázoví špecialisti, ktorí si mysleli, že musí existovať lepší a efektívnejší spôsob ako spracovávať a triediť veľké množstvo dát. Ako sa zlepšovali možnosti využiteľnosti databáz, tak sa zvyšovali aj nároky užívateľov na tieto systémy. Toto viedlo k vytvoreniu v súčasnosti najpoužívanejšieho databázového modelu – k relačnému modelu. (4)

Network Model



Obrázok 4 Sieťový databázový model

4.3.3 Relačný model

Relačný model je najrozšírenejším spôsob uloženia dát v databáze. Jedná sa o spôsob uloženia v logickom zmysle.

Relačná databáza bola prvý krát predstavená v roku 1969. Jej navrhovateľom je Dr. Edgar F. Codd – výskumník IBM, ktorý sa v 60. Rokoch zaoberal novými metódami uloženia veľkého množstva dát. Keďže bol nespokojný s vtedy dostupnými databázovými modelmi, pokúsil sa vyriešiť celé spektrum problémov pomocou matematických techník a štruktúr.

Dr. Codd svoju prácu oficiálne predstavil v roku 1970 kedy vydal prácu s názvom: „Relačný model dát pre veľké zdieľané databanky“¹. Svoj model založil na dvoch matematických disciplínach – teória množín a predikátová logika. Veľmi rozšíreným omylom je, že sa relačný model volá podľa vzťahu medzi dátami. Pravdou

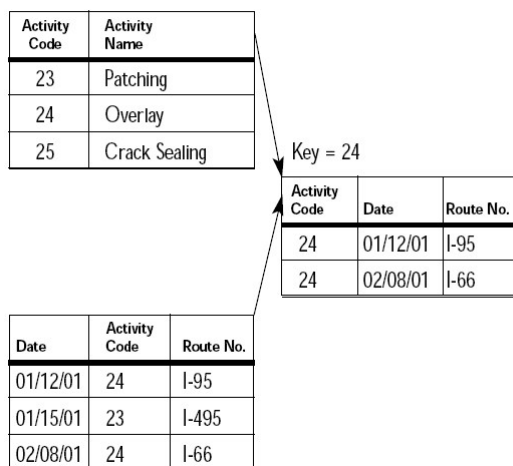
¹ Codd, F. E. *A Relational Model of Data for Large Shared Databanks*. Communications of the ACM, 1970.

však je, že názov je odvodený od matematických relácií na ktorých je model založený.

Relačný databázový model združuje dáta do takzvaných relácií – laicky tabuliek, ktoré obsahujú n-tice (riadky tabuľky). Relácie tvoria základ relačnej databáze. Relácia je štruktúra záznamov s pevne stanovenými položkami (stĺpcami / atribútmi). Skutočné usporiadanie záznamov v databáze je úplne nepodstatné a každý záznam v tabuľke je identifikovaný atribútom, ktorý obsahuje unikátnu hodnotu. Naproti hierarchickému či sieťovému databázovému modelu, užívateľ nemusí poznať fyzické umiestnenie záznamu pokiaľ z neho chce získať určité dáta.

Ak sú v rôznych reláciách (tabuľkách) stĺpce rovnakého typu, potom tieto stĺpce môžu vytvárať väzby medzi jednotlivými tabuľkami. Kolekcia viacerých tabuliek, ich vzťahov a ďalších súčastí potom tvorí relačnú databázu. (4)

Relational Model



Obrázok 5 Relačný dátový model

Relačný model rozlišuje medzi tabuľkami vzťahy **1:1**, **1:N** a **M:N**.

Dvojica relácii (tabuliek) je spolu viazaná vzťahom **1:1** ak je každý záznam z jednej tabuľky zviazaný práve s jedným záznamom v druhej tabuľke. Vzťah typu **M:N** sa väčšinou používa pri podmnožinových tabuľkách. Používa sa tiež napríklad vtedy keď chceme jednotlivým atribútom relácie dať rôzne práve. V tomto prípade jednu tabuľku môžeme rozdeliť na dve vo vzťahu **1:1** a pre jednotlivým užívateľom nastavíme rôzne práva.

Vzťah typu **1:N** sa používa v prípade, keď jeden záznam z prvej tabuľky môže byť viazaný na jeden alebo viac záznamov z druhej tabuľky, ale každý záznam z druhej tabuľky môže byť naviazaný iba na jeden záznam z prvej tabuľky. Vzťah 1:N je pravdepodobne najčastejšie používaný typ vzťahu. (Za jeden tím môže hrať niekoľko hokejistov, ale jeden hokejista môže mať hrať len za jeden tím.)

Dve tabuľky sú prepojené vzťahom **M:N** vtedy, keď jeden záznam z prvej tabuľky môže byť prepojený s jedným alebo viacerými záznamami v druhej tabuľke a zároveň jeden záznam z druhej tabuľky môže byť zviazaný s jedným alebo viacerými záznamami druhej tabuľky. Niekedy je zložitejšie takýto vzťah určiť a preto je vhodné dôkladné preskúmanie tabuliek. (Jeden študent navštevuje v škole niekoľko premetov a zároveň jeden predmet môže navštevovať niekoľko študentov.)

Keďže nie je potrebné poznať fyzické umiestnenie záznamov pri používaní relačného databázového modelu, logicky sa ponúka otázka, ako hľadané dáta z databázy získame. Na získavanie dát slúži jazyk SQL – štruktúrovaný dotazovací jazyk (structured query language). Jazyk SQL obsahuje príkazy na kompletnú správu relačnej databázy – nie len na čítanie, ale aj zápis, editáciu, vytváranie nových relácií, spravovanie užívateľov a podobne. (4)

```
SELECT reg_hotel.Name, Count(*) AS Pocet
FROM trl_order
Inner Join reg_hotel ON trl_order.HotelId = reg_hotel.Id
WHERE trl_order.State = 4 AND trl_order.TotalPrice <> 0
GROUP BY reg_hotel.Name
ORDER BY Pocet DESC
```

Obrázok 6 Príklad SQL dotazu

4.3.4 Ostatné modely

Napriek tomu, že sa vo väčšine prípadov používa práve relačný databázový model, okrem vyššie spomínaného hierarchického a sieťového databázového modelu existoval aj plochý databázový model a v súčasnosti sa využíva tiež dimenzionálny databázový model používaný hlavne pri OLAP databázach alebo v posledných rokoch aj objektovo orientovaný databázový model. (4)

4.4 Proces tvorenia databázy

Na to aby mohla akákoľvek databáza fungovať, musí byť v prvom rade niekým vytvorená. Proces tvorby sa dá podľa Hernandeza rozdeliť na 3 základne kroky:

1. Logický návrh
2. Fyzická implementácia
3. Vývoj aplikácie

Prvá fáza sa zoberá určením a definovaním relácií, ich atribút, vzťahov, stanovanie primárnych cudzích kľúčov a zaistením integrity dát na rôznych úrovniach. Už v tejto fáze by mal byť návrh čo najviac kompletný.

Fyzická implementácia prináša samotné vytváranie tabuliek v ktorých budú dáta uložené. V tejto fáze niekedy nastáva problém a implementácia na rôznych databázových systémoch sa môže líšiť kvôli obmedzeniam, alebo kvôli zabezpečeniu vyššieho výkonu. Fyzicky implementovaná databáza môže mať odlišnú štruktúru ako logický návrh, ale mala by obsahovať všetky navrhované atribúty relácií z logického návrhu.

Posledná fáza – vývoj aplikácie – hovorí o vytvorení aplikácie, ktorá bude stáť „nad“ navrhovanou databázou a umožní užívateľom pracovať s dátami uloženými v databáze.

Všetky fázy procesy tvorenia databázy sú rovnako dôležité a je potrebné venovať im dostatočnú pozornosť. Bez logického návrhu by nemohla vzniknúť fyzická implementácia a bez nej by nemohla existovať aplikácia ktorá by umožňovala pracovať s dátami uloženými v databáze. (4)

4.4.1 Tradičné metódy logického návrhu

Logický návrh ako prvý krok tvorby databázy môžeme ďalej rozdeliť na tri fázy. V prvom rade je potrebné dôkladne **analyzovať požiadavky** na systém, potom uskutočniť fázu **dátového modelovania** a nakoniec určitým spôsobom navrhnutý model **normalizovať**.

V prvej fáze – analýza požiadaviek – sú zahrnuté činnosti ako skúmanie modelovaného podniku. Rozhovory s budúcimi užívateľmi, respektíve manažmentom, ktoré by mali smerovať k zhodnoteniu súčasnej situácie. V tejto fáze by sme mali byť schopný určiť čo konkrétne daný podnik potrebuje, na čo to bude využívať a v akom rozsahu.

Vo fáze modelovania dát používame metódy na to určené, ako napríklad diagram vzťahov medzi entitami (entity-relationship diagram – ER diagram), sémantické modelovanie objektov, alebo modelovanie objektov podľa ich úlohy. Každá z týchto metód modelovania obsahuje spôsob ako vizuálne interpretovať rôzne aspekty databázovej štruktúry ako sú tabuľky, vzťahy medzi nimi a charakteristiky vzťahov.

V priebehu druhej fázy je potrebné k navrhovaným tabuľkám priradiť ich polia (atribúty). Každdej tabuľke je tiež priradený primárny kľúč, sú identifikované a implementované rôzne úrovne integrity dát a vytvorené vzťahy pomocou cudzích kľúčov. Keď sú všetky tieto kroky vykonané, nastáva fáza normalizácie.

Pri procese normalizácie dochádza k odstraňovaniu nedostatkov navrhovaných tabuliek ako sú redundancia dát alebo možnosti vzniku aktualizáčnej anomálie. V tomto procese nastáva rozklad veľkých tabuliek na menšie. Postup normalizácie je rozdelený do niekoľkých krokov, pričom po dokončení každého z nich sa tabuľka nachádza v určitej normálnej forme. (4)

4.4.2 Normálne formy

Normálna forma je špecifická množina pravidiel, ktoré sa používajú na testovanie či je štruktúra určitej tabuľky bezproblémová a bezchybná. Existuje niekoľko normálnych foriem a každá z nich sa využíva k testovaniu určitých špecifických problémov. V súčasnosti sa používa prvá normálna forma, druhá normálna forma, tretia normálna forma, štvrtá normálna forma, piata normálna forma, Boyce-Coddova normálna forma a doménovo-kľúčová normálna forma.

Prvé tri normálne formy by mali byť splnené vždy a preto uvádzam ich charakteristiku:

Relácia (tabuľka) je v prvej normálnej forme, ak sú všetky jej atribúty (polia) definované nad skalárnymi obormi hodnôt (doménami). To znamená, že tabuľka neobsahuje žiadne zložené alebo viachodnotové atribúty.

Relácia je v druhej normálnej forme, ak je v prvej normálnej forme a navyše všetky jej atribúty sú funkčne závislé na celkom kandidátnom kľúči – nie len na časti kandidátneho kľúča.

Relácia je v tretej normálnej forme ak je v druhej normálnej forme a navyše všetky jej kľúčové atribúty sú vzájomne funkčne nezávislé. Takže položka, ktorá nie je kľúčová, nesmie byť závislá na inej kľúčovej položke. (5)

4.5 Technológie

4.5.1 MySQL

MySQL je databázový systém, vytvorený švédskou firmou MySQL AB. Je k dispozícii pod bezplatnou licenciou GPL, ako aj pod komerčnou platenou licenciou.

MySQL je multiplatformá databáza. Komunikácie prebieha pomocou jazyka SQL. Podobne ako u iných SQL databáz sa jedná o dialekt tohto jazyka s niektorými rozšíreniami.

Pre svoju jednoduchú implementovateľnosť (dá sa inštalovať na Linux, MS Windows, ale i ďalšie operačné systémy), výkon a predovšetkým vďaka tomu, že sa jedná o voľne šíriteľný software, má vysoký podiel na v súčasnej dobe používaných databázach. (13)

4.5.2 IIS

Internet Information Services (IIS) znamená v preklade Internetová Informačná Služba. Je to súbor internetových služieb pre servery od spoločnosti Microsoft pre operačné systémy spoločnosti Microsoft.

IIS je druhou najpoužívanejšou platformou pre realizáciu webového servera a v súčasnosti podporuje tiež FTP, FTPS, SMTP, NNTP a HTTP/HTTPS protokoly. (10)

4.5.3 ASP.NET

ASP.NET je framework² pre tvorbu webových aplikácií od spoločnosti Microsoft. Pomocou neho majú programátori možnosti budovať dynamické web stránky, webové aplikácie alebo webové služby. ASP.NET je postavená na „Common Language Runtime“ (CLR), čo umožňuje programátorom písať ASP.NET kód v ktoromkoľvek s podporovaných jazykov. (14)

² Framework je softwarová štruktúra, ktorá slúži ako podpora pri programovaní a vývoji a organizácii iných softwarových projektov. Môže obsahovať podporné programy, knižnicu API, návrhové vzory alebo doporučené postupy pri vývoji.

4.6 Diagramy

4.6.1 ER diagram

ER model je založený na chápaní sveta ako množiny základných objektov – entít (Entity) a vzťahov (Relationship) medzi nimi. Popisuje dáta "v pokoji", neukazuje, aké operácie s dátami budú prebiehať.

Pri modelovaní ER diagramu by mali byť mená zrozumiteľná. Entitné množiny by mali niesť podstatné meno a vzťahové množiny slovesá alebo predložky. Pri tvorbe diagramov sa používajú rôzne grafické prvky. Neexistuje však žiadna norma ISO, ktorá by upravovala značenia ER diagramov. (11)

4.6.2 EPC diagram

EPC diagram (z anglického Event-driven Process Chain, diagram procesu riadeného udalosťami) je metóda používaná pri modelovaní procesov. Pomocou diagramu sa dá prehľadne definovať, akými aktivitami bude proces realizovaný, v akom poradí a ako budú jednotlivé aktivity koordinované. (15)

EPC diagram sa skladá z nasledujúcich elementov:

- Aktivity - základný kameň určuje čo má byť vykonané
- Udalosti – popisujú situácie pred a po vykonaní určitej aktivity jednotlivé aktivity sú vzájomne prepojené pomocou udalostí. Udalosť predstavuje výstupnú podmienku jednej aktivity a zároveň vstupnú podmienku inej aktivity
- Logické spojky - používajú sa pre prepojovanie jednotlivých aktivít a udalostí

4.6.3 Vývojový diagram - Flowchart

Vývojový diagram je grafické znázornenie určitého algoritmu alebo procesu. Vývojový diagram používa pre znázornenie jednotlivých operácií symboly, ktoré sú navzájom prepojené pomocou orientovaných šípok. (12)

- úsečka či množina nadväzujúcich úsečiek končiaca šípkou určuje smer spracovania
- obdĺžnik s popisom definuje jednotlivé kroky spracovania
- kosoštvorec slúži na vetvenie v závislosti na splnení podmienky
- obdĺžnik so zaoblenými rohmi znázorňuje začiatok alebo koniec algoritmu
- kruh - spojka, jednotlivých úsečiek

5 Vlastné návrhy riešenia a ich prínosy

Na základe analýzy problému a súčasného stavu, som bol schopný identifikovať nároky na systém a teda aj nároky na databázu pre tento informačný systém.

V priebehu písania tejto práce cestovná kancelária Solartour schválila mnou navrhované postupy riešenia a preto v tejto kapitole budem popisovať to, ako som v skutočnosti pri realizácii tohto projektu postupoval.

5.1 Časový harmonogram

Časový plán bol postavený tak, aby bolo možné spustiť testovaciu prevádzku nového systému najneskôr do začiatku výstavy cestovného ruchu – SLOVAKIATOUR 2009, teda do 22. 1. 2009. Všetky ostatné činnosti podliehali tomuto termínu.

Časový plán je zostavený tak, aby jednotlivé činnosti na seba nadväzovali a zbytočne nevznikali prázdne časové intervaly, počas ktorých by sa na projekte nepracovalo.

Poradie	Názov činnosť	Dátum ukončenia
1.	Zoznámenie sa s fungovaním cestovnej kancelárie a stanovovanie cieľov	30. 9. 2008
2.	Analýza súčasného stavu	15. 10. 2009
3.	Stanovenie požiadavkov na systém	30. 10. 2009
4.	Návrh logickej schémy budúcej databáze a odsúhlasenie cestovnou kanceláriou	15. 11. 2008
5.	Fyzická implementácia databáze	30. 11. 2008
6.	Vývoj webovej prezentácie a administrácie	22. 1. 2009
7.	Testovacia prevádzka	30. 4. 2009
8.	Implementácia navrhovaných zmien	30. 5. 2009
9.	Ostrá prevádzka	

Tabuľka 1 Časový harmonogram

Prvou fázou tohto projektu bolo zoznámenie sa s fungovaním cestovnej kancelárie a s procesmi, ktoré prebiehajú v jej vnútri. Po krátkej analýze bolo možné skonštatovať, že vytvorenie informačného systému pre potreby on-line predaja je realizovateľné. V tejto fáze boli stanovené základné ciele, ktoré by mal projekt splniť.

Po tejto fáze nasledovala v poradí druhá fáza – analýza súčasného stavu. Táto fáza bola pre projekt veľmi dôležitá, pretože je nutné identifikovať jednotlivé procesy

a je potrebné zistiť, kde sa nachádzajú slabé stránky, ktoré bude môcť nový informačný systém odstrániť.

Výstup z tejto fázy slúžil ako podklad k stanoveniu požiadavkou na nový systém. Pri tejto analýze som problém rozdelil na dve kategórie. Čo chce cestovná kancelária prostredníctvom svojej web stránky ponúkať zákazníkom a na druhej strane, čo od informačného systému očakávajú jej zamestnanci. Po skombinovaní týchto dvoch pohľadov som dospel k viac či menej ucelenej predstave čo všetko musí databázová štruktúra obsahovať.

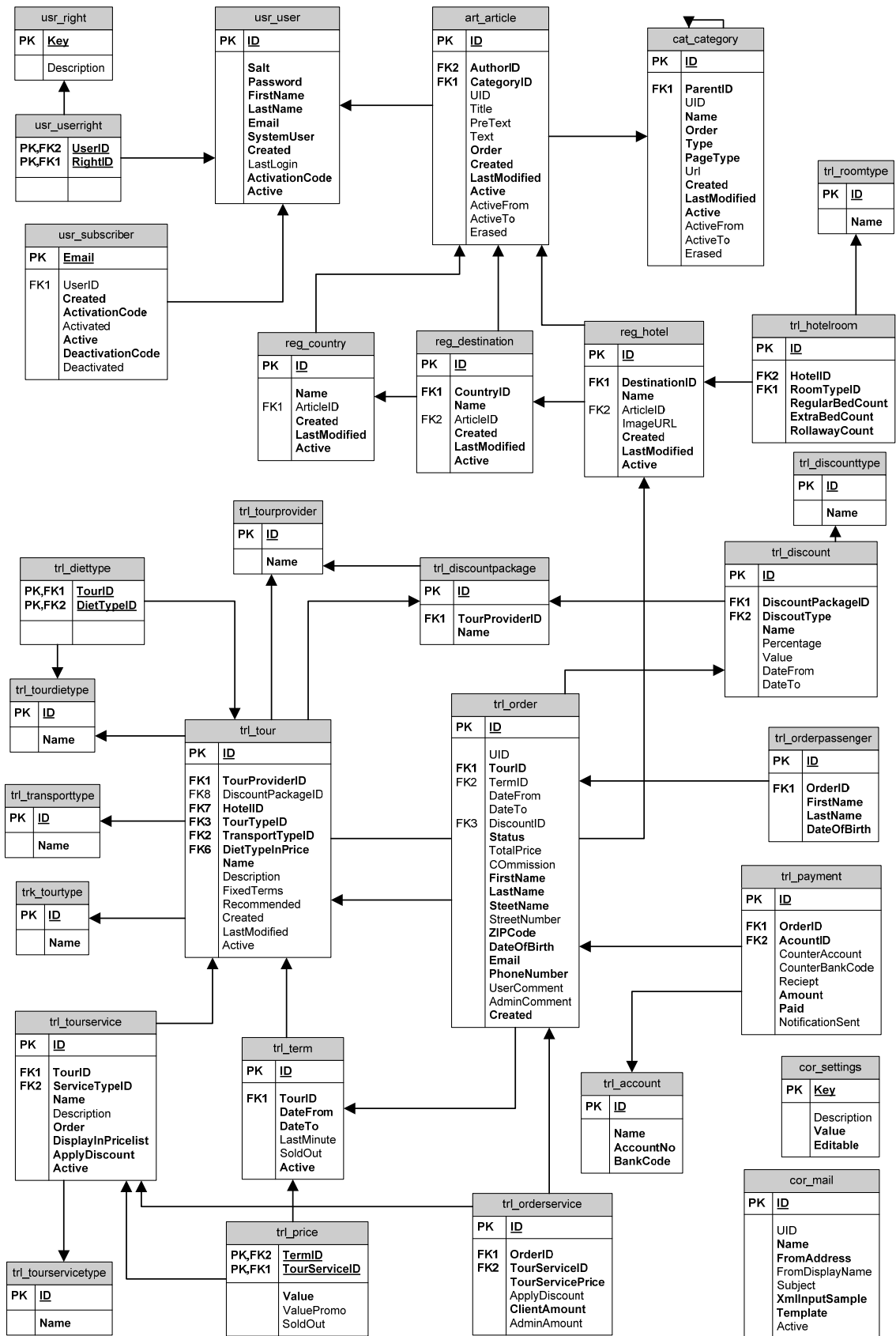
V ďalšej fáze bolo potrebné identifikované požiadavky preniesť do logického návrhu databázy. Bolo potrebné identifikovať jednotlivé entity, stanoviť medzi nimi väzby a vypísať všetky atribúty, ktoré budú nové aplikácie potrebovať. Táto fáza bola jedna z najdôležitejších, pretože výstup z tejto fázy mala schváliť cestovná kancelária a na základe vypracovaného dokumentu bola realizovaná fyzická implementácia databázy ako aj vytvorenie webovej prezentácie a administratívneho rozhrania.

Táto bakalárska práca má za úlohu návrh databázovej štruktúry a preto ďalšie kroky realizácie projektu nebudem podrobne vysvetľovať. Po fyzickej implementácii nastala najdlhšia fáza - vývoj webovej prezentácie a administrácie. Informačný systém a nová webová prezentácia bola pripravená k testovacej prevádzke v dohodnutom termíne. Momentálne sa systém nachádza vo fáze testovania a fáze analýzy ďalších požiadaviek na systém.

Na základne skúseností s predajom počas prvých mesiacov bude vypracovaná analýza toho, čo je potrebné v systéme zmeniť, alebo doplniť. Po implementácii požadovaných zmien bude možné nasadiť informačný systém s webovou prezentáciou do ostrej prevádzky.

Aj počas ostrej prevádzky sa samozrejme vyskytnú problémy, respektíve situácie s ktorými sa vopred nepočítalo a teda ak sa cestovná kancelária rozhodne informačný systém dlhodobo využívať, predpokladám, že sa tento systém bude neustále vyvíjať, zlepšovať, alebo len prispôbovať meniacim sa požiadavkám zo strany zadávateľa, teda cestovnej kancelárie Solartour.

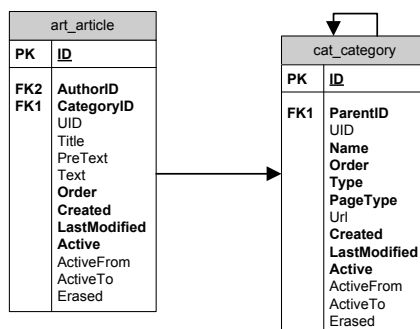
5.2 Logický návrh



Obrázok 7 Logický návrh databázovej štruktúry

5.2.1 Textové dáta

Pri návrhu databázovej štruktúry som bral ohľad na to, že bude potrebné ukladať veľké množstvo textových informácií. Tieto informácie je potrebné triediť a filtrovať. Pre uchovávanie a štrukturovanie všetkých textov, ktoré budú administrátorom editovateľné slúžia dve tabuľky: *art_article* a *cat_category*.



Obrázok 8 Schéma tabuliek pre textové dáta

Tabuľka *cat_category* vytvára určitú stromovú štruktúru webovej prezentácie, a v jednotlivých článkoch v tabuľke *art_article* sú uložené konkrétne texty, ktoré sa určitým spôsobom budú zobrazovať na stránke.

Spoločné atribúty pre viacero tabuliek v tejto logickej schéme sú:

- **UID** (Unique Identifier) – predstavuje unikátny reťazec znakov, slúžiaci pre tvorenie linkov vyhovujúcim pravidlám pre optimalizáciu pre vyhľadávače
- **Order** – používa sa vždy keď má možnosť administrátor ovplyvniť poradie zobrazovania určitých prvkov
- **Created** – Predstavuje dátum a čas vytvorenia prvku
- **LastModified** – Posledná zmena položky – dátum a čas
- **Active** – Určuje či je daná položka aktívna (či sa zobrazuje)
- **Erased** – Vyjadruje či bola položka vymazaná (kvôli možným neúmyselným zásahom do systému aj vymazané položky zostávajú v databáze)

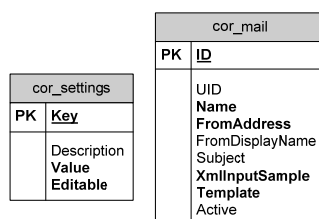
Jednotlivé články okrem textu a spoločných atribútov majú atribúty pre stanovanie Názvu/Nadpisu, Anotácie (PreText) a Dátumu pre začiatok a koniec platnosti článku.

Kategórie, ktoré tvoria základnú štruktúru pre webovú prezentáciu, obsahujú okrem informácie o ich názve aj atribút hovoriaci o type kategórie. Typom môže byť

napríklad jednoduchý výpis článkov, ktoré patria do tejto kategórie. Ďalšími typmi môžu byť napríklad externý odkaz na lokalitu na internete, interný odkaz alebo vopred definovaná zostava³. Na uchovávanie informácií o type stránky a jej smerovaní slúžia atribúty Type, PageType a Url. Týmto spôsobom je možné vytvoriť a administrovať celú navigáciu po webovej prezentácii.

5.2.2 Nastavenia systému

Z dôvodu potreby ukladania rôznych systémových údajov sa v databázovej štruktúre nachádzajú dve tabuľky. Jedná sa o: *cor_mail* a *cor_settings*.



Obrázok 9 Schéma tabuliek pre nastavenie systému

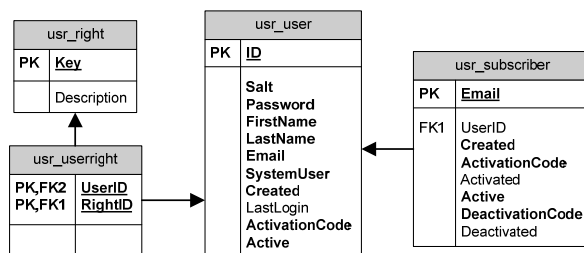
Tabuľka *cor_settings* obsahuje definície premenných, ktoré sa budú ďalej používať v administrácii a vo webovej prezentácii. Takýmito premennými sú napríklad dĺžka cachovania stránok, relatívna cesta k adresáru s obrázkami, príznak či sa majú novo zaregistrovaní užívatelia automaticky aktivovať a podobne.

Keďže systém má byť pripravený na rozposielanie automatických mailov rôzneho druhu, druhou systémovou tabuľkou je tabuľka *cor_mail*, ktorá definuje názov hromadne rozposielaného mailu, od koho má byť e-mail poslaný, s akým predmetom, a aká šablóna sa používa na formátovanie tohto mailu. Atribút *XmlInputSample* obsahuje vzorové dáta pre potreby testovania a editovania šablóny mailu.

5.2.3 Riadenie prístupu a správa užívateľov

Práva k prístupu, editácii či mazaniu záznamov riešia tabuľky: *usr_right*, *usr_userright*, *usr_user*.

³ Zostavou sa rozumie určitým spôsobom formátované dáta zobrazené pre návštevníka stránky. Príkladom zostavy môže byť stránka s informáciami o hoteli, kde sa nachádza článok o hoteli spolu s cenníkom v tabuľkovej forme, popisom cenníka, cenovou kalkulačkou a formulárom na predbežnú objednávku.



Obrázok 10 Schéma tabuliek pre správu užívateľov

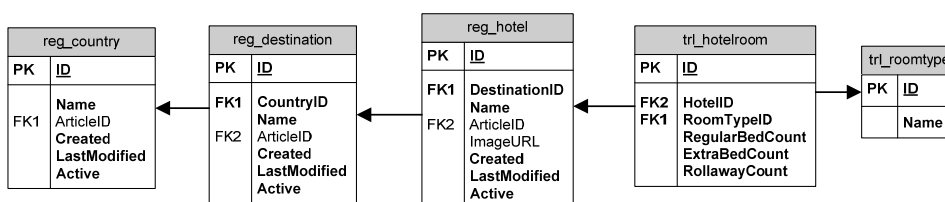
Prístup k administrácii a k citlivým údajom o klientoch a objednávkach je potrebné chrániť. Z tohto dôvodu sú v databázovej štruktúre pripravené tabuľky na ukladanie informácií o administrátorských účtoch a o jednotlivých právach užívateľov.

Pre potreby prihlasovania sa k odberu noviniek návštevníkmi stránky je pripravená tabuľka *usr_subscriber*. Momentálne sa návštevníci stránky nemusia registrovať a teda vytvárať si vlastné účty kvôli zaslaníu predbežnej objednávky. Tabuľky sú však navrhnuté tak, aby to bolo v budúcnosti možné a emailová adresa zapísaná do mailing-listu bude môcť byť priradená ku konkrétnemu užívateľovi.

5.2.4 Regióny

Pretože jednotlivé zájazdy treba odlišovať aj geograficky, databázová štruktúra je pripravená na triedenie zájazdov v troch vrstvách. Na najnižšom mieste je hotel, ktorý patrí do nejakého letoviska a letovisko patrí do určitej krajiny.

K tejto štruktúre sa pripája sa definícia typov ubytovania pre konkrétny hotel.

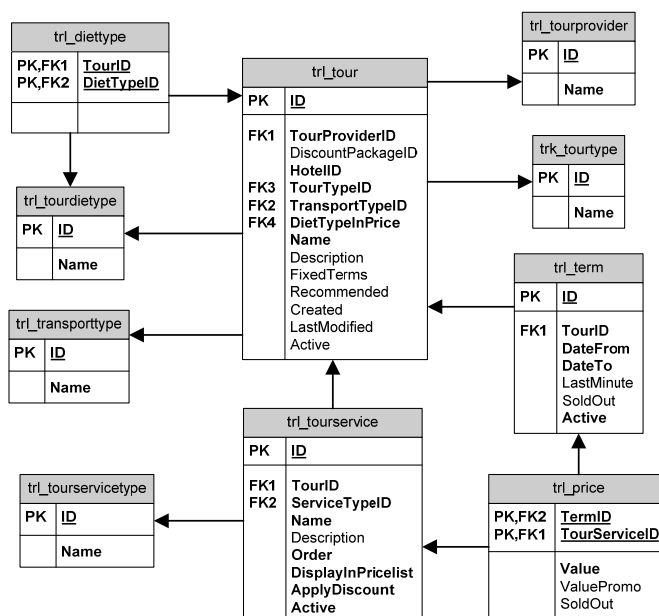


Obrázok 11 Schéma tabuliek pre regionálnu správu

Ku krajine, letovisku aj hotelu je možné priradiť článok. Tieto texty môžu byť neskôr použité pri rôznych zostavách. Napríklad pri zostave hotelov jedného letoviska. Aby som zabezpečil možnosť filtrovania cez rôzne typy ubytovania, v databázovej štruktúre sa nachádzajú tabuľky *trl_hotelroom* a *trl_roomtype*. Na základe ich atribútov bude možné implementovať toto filtrovanie. V tabuľke *trl_hotelroom* sa nachádzajú počty rôznych typov lôžok v izbe. Tak bude možné odlíšiť aký je minimálny a maximálny počet pevných lôžok a koľko prísteliek je možné do izby pridať.

5.2.5 Modul správy zájazdov

V regionálnej štruktúre stále chýba jedna podstatná entita, ktorou je samotný zájazd. Ako som vysvetlil vyššie, zájazd by sa dal chápať ako „cenník“ určitého hotela s určitou dopravou pre určité termíny. Pre jeden hotel môže byť definovaných viacero zájazdov.



Obrázok 12 Schéma tabuliek pre správu zájazdov

Správu zájazdov riešia štyri hlavné tabuľky: *trl_tour*, *trl_terms*, *trl_tourservice* a *trl_price*. V tabuľke *trl_tour* sú uložené informácie o hoteli, do ktorého ide konkrétny zájazd, kto zájazd organizuje, či je poskytnutá na zájazd nejaká zľava first minute, o aký typ zájazdu sa jedná, aká je doprava, aké sú možnosti stravovania a aká strava je zahrnutá v cene, či sú turnusy zájazdu pevne dané alebo si zákazník si môže sám vybrať začiatok aj koniec svojej dovolenky.

Cenník zájazdu tvoria termíny (turnusy) na jednej strane a služby⁴ na druhej strane. Dostaneme tak tabuľku dostupných služieb a termínov pre konkrétny zájazd. Pri termínoch v tabuľke *trl_term* môžeme okrem termínom nastaviť príznaky či sa jedná o termín Last Minute a teda bude určitým spôsobom zvýraznený, alebo je termín kompletne vypredaný a už sa teda prostredníctvom web stránky nebude dať objednať. V tabuľke *trl_service* môže administrátor okrem názvu (tiež popisu) a typu služby

⁴ Službou rozumiem čokoľvek čo cestovná kancelária, alebo jej zmluvní partneri poskytujú zákazníčkovi. Medzi základné služby v cestovnom ruchu radíme rôzne druhy ubytovania, medzi príplatkové potom napríklad dopravu, stravu, poistenie a podobné...

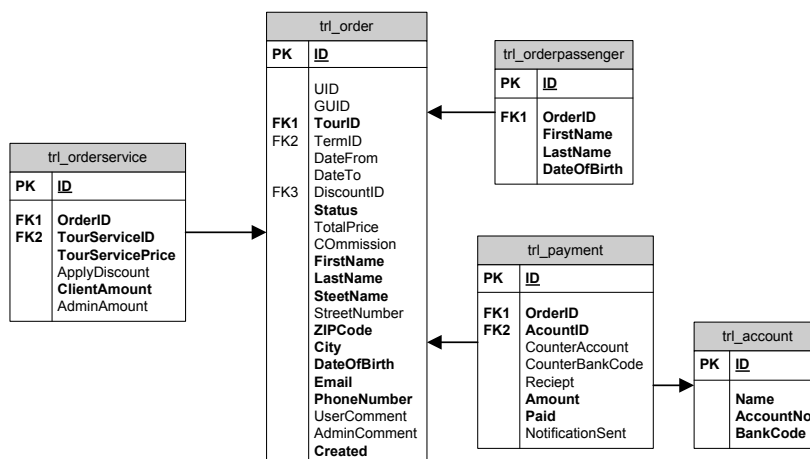
ovplyvňovať poradie, či sa služba bude zobrazovať v tabuľkovej forme cenníku a či sa na konkrétnu službu aplikuje zľava first minute.

Ceny v jednotlivých bunkách v tomto pomyselnom cenníku sa ukladajú do tabuľky *trl_price*. Tak ako môže byť vypredaný celý termín, môže byť vypredaná aj konkrétna služba v termíne. Pre potreby vytvorenia akciovej (Last minute, resp. iné akcie) ceny určitej služby je v tabuľke *trl_price* pripravený atribút ValuePromo.

Ostatné tabuľky v tomto module sú číselníky a slúžia na podporu hlavných tabuliek. V budúcnosti by podľa potreby mohli byť pridané rôzne atribúty, ale podľa požiadaviek cestovnej kancelárie to momentálne nie je potrebné.

5.2.6 Modul správy objednávok

Pre potreby uchovávanie informácií o objednávkach na diaľku slúži hlavná tabuľka *trl_order* a tabuľky *trl_orderpassenger* a *trl_orderservice*. Tabuľky *trl_payment* a *trl_acount* slúžia na uchovávanie informácií o platbách.



Obrázok 13 Schéma tabuliek pre správu objednávok

Pri objednávke potrebuje vedieť cestovná kancelária niektoré základné informácie o objednávateľovi a jeho spolucestujúcich. Informácie o objednávateľovi ako jeho Meno, Priezvisko, Adresa (ulica, popisné číslo a smerové číslo a mesto), Dátum narodenia a kontaktné údaje (Email a Telefón) sa ukladajú do tabuľky *trl_order* spolu s informáciami o aký zájazd v akom termíne má zákazník záujem. Ďalej je v tejto tabuľke uložená informácia o zľave platnej v čase objednania. Pre štatistické potreby môže administrátor vyplniť konečnú predajnú cenu a províziu, ktorú by cestovná

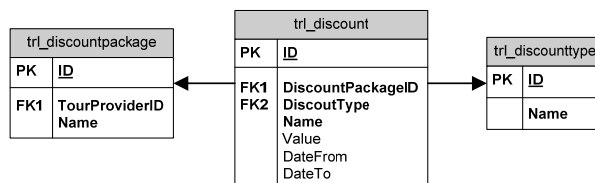
kancelária poskytla províznemu predajcovi⁵. Odosielateľ objednávky ako aj administrátor majú možnosť k objednávke pripojiť krátku poznámku. Administrátor môže objednávke prideliť vlastné UID – napríklad číslo zmluvy. GUID predstavuje náhodne vygenerovaný reťazec, ktorý bude súčasťou URL stránky s možnosťou sledovania objednávky pre zákazníka.

K objednávke sú ďalej pripojené tabuľky *trl_orderservice* a *trl_orderpassenger*. Tabuľka *trl_orderservice* obsahuje dáta o objednaných službách, ich cenách, objednanom množstve a o tom, či je na objednanú službu poskytnutá zľava. Tabuľka *trl_orderpassenger* je navrhnutá tak, aby bolo možné uchovávať informácie o spolucestujúcich – Meno, Priezvisko a Dátum narodenia.

Tabuľky *trl_account* a *trl_payment* evidujú platby k jednotlivým objednávkam. V tabuľke *trl_account* sú uložené informácie o jednotlivých účtoch na ktoré je možné zájazd uhradiť. Účtom v tomto prípade môže byť aj hotovostná pokladňa. Tabuľka *trl_payment* uchováva dáta o jednotlivých platbách za konkrétne objednávky. Bežnou praxou je platba zálohy pri podpise zmluvy a následný doplatok určitý čas pred odchodom na zájazd. Administrátor okrem zaplatenej sumy, dátumu určuje aj na aký účet bola platba prijatá a odkiaľ platba prišla. Má tiež možnosť zaznamenávať, kde sa dá platba dohľadať (číslo výpisu, ...) a či bola zákazníkovi odoslaná informácia o prijatí platby.

5.2.7 Zľavy

Počas analýzy problému a stanovovania požiadaviek na systém bola identifikovaná potreba určitým spôsobom evidovať zľavy, respektíve balíky zliav. Preto boli do návrhu databázovej schémy zahrnuté tabuľky *trl_discountpackage*, *trl_discount* a *trl_discounttype*.



Obrázok 14 Schéma tabuliek pre správu zliav

⁵ Ak sa jedná o zájazd organizovaný inou cestovnou kanceláriou, administrátor môže vyplniť províziu, akú od tejto cestovnej kancelárie dostal.

Informácie o samotnej zľave budú uložené v tabuľke *trl_discount*. Tu je špecifikovaný názov zľavy, typ zľavy (buď percentuálna, alebo absolútna), jej hodnota a časový interval od kedy do kedy je zľava platná.

Jednotlivé zľavy je potrebné združovať do balíkov a až tieto balíky aplikovať na konkrétne zájazdy. Balík zliav „Zľavy za skorý nákup na autobusové zájazdy“, môže potom obsahovať rôzne zľavy pre jednotlivé mesiace v roku.

5.3 Prínosy pre cestovnú kanceláriu

Zavedenie informačného systému a vytvorenie novej webovej stránky bolo určite pre cestovnú kanceláriu prínosné. Jednotlivé prínosy môžeme rozdeliť do dvoch skupín: merateľné a nemerateľné.

Nemerateľné prínosy síce nie je možné vyčíslit' v peniazoch, ale je možné ich identifikovať. Jedná sa napríklad o:

- Možnosť vytvorenie objednávky zákazníkom kedykoľvek a kdekoľvek prostredníctvom internetu
- Oslovenie širšieho okruhu potenciálnych klientov keďže cestovná kancelária Solartour nemá obchodné zastúpenie vo všetkých mestách na Slovensku.
- Odbúranie nepotrebnéj a niekedy zdĺhavej komunikácie medzi cestovnou kanceláriou a zákazníkmi
- Zvýšenie „prestíže“ vďaka modernej a funkčnej webovej prezentácii
- Zlepšenie celkového povedomia o cestovej kancelárii, keďže svoje služby ponúka aj priamo a nie len prostredníctvom provízií predajcov

Aj keď je systém už zavedený a momentálne sa nachádza v testovej fáze, zatiaľ nie je možné exaktne prezentovať merateľné prínosy pre cestovnú kanceláriu za celú sezónu. Ak by sme však zobrali údaje, ktoré sú dostupné ku koncu apríla 2009, môžeme porovnať určité hodnoty aj počas rozbehnutej sezóny:

- Oproti minulému roku nastalo **zvýšenie počtu objednávok** v období do konca mesiaca apríl o **482%**.
- Napriek tomu, že sú zatiaľ k dispozícii **výsledky predaja len za približne prvú polovicu sezóny**, počet objednávok k 30. 4. 2009 už **presiahol 2,9-násobok počtu objednávok za celú minulú sezónu**.

6 Záver

Cieľom mojej bakalárskej práce bolo navrhnúť dátovú štruktúru na základe ktorej bude možné vytvoriť on-line informačný systém na správu objednávok, zájazdov a služieb v jednotlivých krajinách, letoviskách a ubytovacích kapacitách. Ďalej z tejto dátovej štruktúry mala vychádzať webová stránka cestovnej kancelárie s možnosťou on-line objednávky, respektíve rezervácie a s možnosťou cenovej kalkulácie kompletnej ceny zájazdu vrátane povinných či nepovinných príplatkov a vrátane odpočítania aktuálnych zliav.

Na základe vykonanej analýzy a po diskusii s predstaviteľmi cestovnej kancelárie bola navrhnutá dátová štruktúra, ktorá splňuje všetky požiadavky pre informačný systém ako aj pre webovú prezentáciu. Na základe tejto dátovej štruktúry boli vytvorené obidve webové aplikácie a momentálne sa nachádzajú v testovacej fáze.

Táto bakalárska práca splnila svoj cieľ a jej výstupy zjednodušili a zefektívnilili prácu cestovnej kancelárie a sprístupnili aktuálne informácie potenciálnym klientom a províznym predajcom.

7 Zoznam použitej literatúry

7.1 Monografické zdroje

- [1] DRUSKA, P. CSS a XHTML : Tvorba dokonalých webových stránok krok za krokom. Grada Publishing, 2006. ISBN 80-247-1382-9.
- [2] ECKER, C. Profesionální webdesign : techniky a vzorová řešení. 1. vyd. Brno: CP Books, 2005. 421 s. ISBN: 80-251-0547-4.
- [3] GROFF, James R. a WEINBERG, Paul N. SQL : Kompletní průvodce. 2005. 936 s. ISBN 80-251-0369-2.
- [4] HERNANDEZ, Michael J. Návrh databází. 2006. 408s. ISBN 80-247-0900-7.
- [5] KOCH, M. Datové a funkční modelování. Brno: CERM, 2006. 108 s. ISBN 80-214-3252-7.
- [6] KOSEK, J. HTML. Tvorba dokonalých WWW stránek. Grada, 1998. ISBN 80-7169-608-0.
- [7] KŘÍŽ, J. a DOSTÁL, P. Databázové systémy : studijní text 1. vyd. Brno : Akademické nakladatelství CERM, s.r.o., 2005. ISBN 80-214-3064-8.
- [8] PALOVSKÁ, Helena. Databáze jako informační zdroj pro uživatele. 2004. 87s. ISBN 80-245-0720-X.
- [9] ZENDULKA, Jaroslav. Databázové systémy. Brno : Fakulta informačních technologií, 2008.

7.2 Elektronické zdroje

- [10] GUTHRIE, S. IIS 7.0. [on-line]. Dostupné z <<http://weblogs.asp.net/scottgu/archive/2007/04/02/iis-7-0.aspx>>. Získané dňa 5. 3. 2009.
- [11] CHAPPLEM, M. Entity-Relationship Diagram [on-line]. Dostupné z <<http://databases.about.com/cs/specificproducts/g/er.htm>>. Získané dňa 5. 5. 2009.
- [12] SEVOCAB: Software and Systems Engineering Vocabulary, Term: Flow chart [on-line]. Dostupné z < http://pascal.computer.org/sev_display/index.action>. Získané dňa 5. 3. 2009.
- [13] SCHUMACHER, R. a LENTZ, A. Dispelling the Myths, MySQL AB. [on-line] Dostupné z <<http://dev.mysql.com/tech-resources/articles/dispelling-the-myths.html>> Získané dňa 5. 3. 2009.

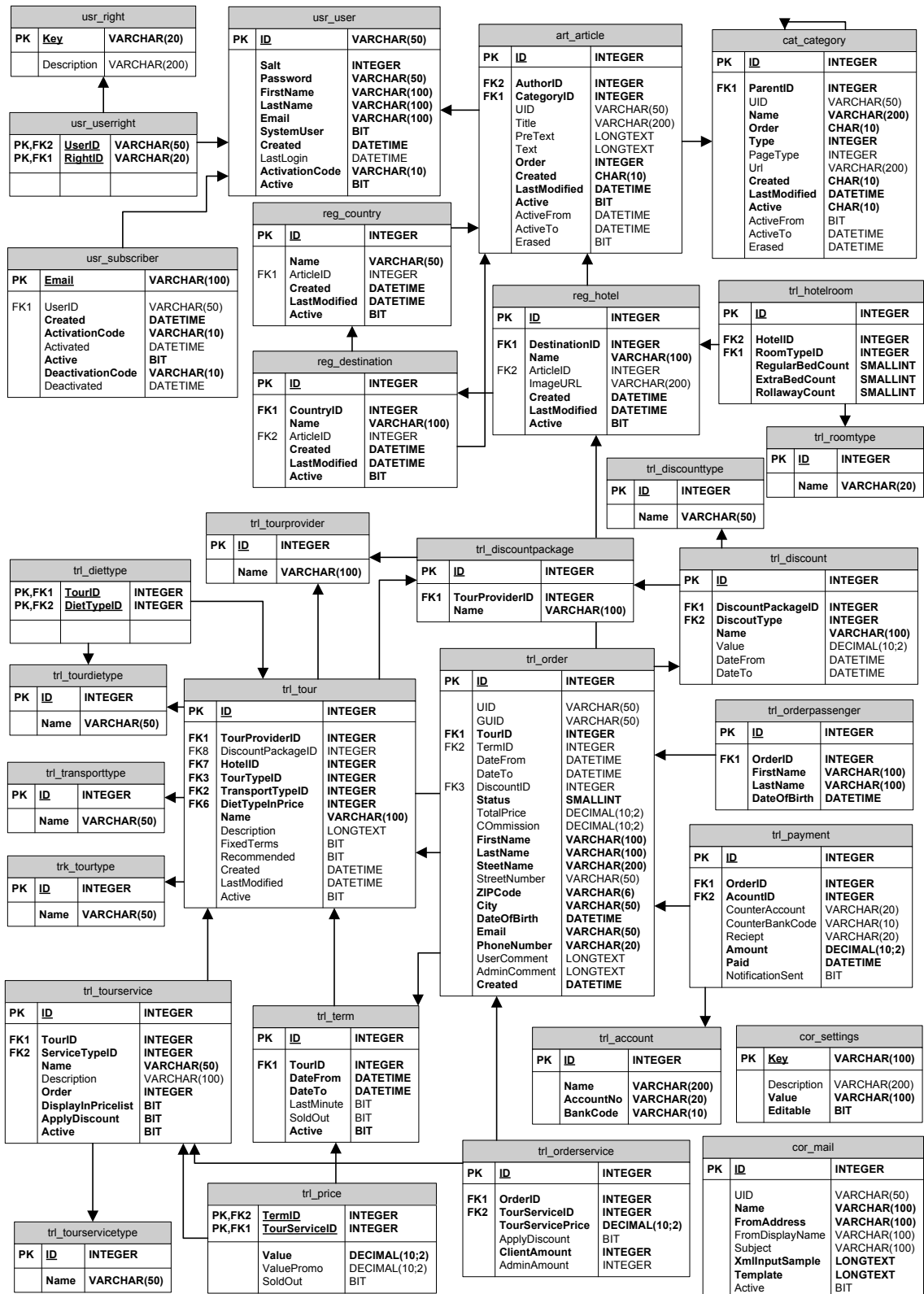
- [14] STANEK, I. ASP.NET Quickstart Tutorial. [on-line]. Dostupné z
<<http://quickstart.aspnet.sk/QUICKSTARTV20/aspnet/doc/default.aspx>>.
Získané dňa 18. 5. 2009.
- [15] VAN DER AALST, W. Formalization and Verification of Event-driven Process
Chains [on-line]. Dostupné z
<<http://is.tm.tue.nl/staff/wvdaalst/publications/p74.pdf>>. Získané dňa 5. 3.
2009.

8 Zoznam obrázkov a tabuliek

Obrázok 1 EPC diagram - očakávané správanie zákazníka na web stránke.....	17
Obrázok 2 Flowchart - Vybavovanie objednávky	19
Obrázok 3 Hierarchický databázový model.....	22
Obrázok 4 Sieťový databázový model.....	23
Obrázok 5 Relačný dátový model.....	24
Obrázok 6 Príklad SQL dotazu	25
Obrázok 7 Logický návrh databázovej štruktúry	32
Obrázok 8 Schéma tabuliek pre textové dáta.....	33
Obrázok 9 Schéma tabuliek pre nastavenie systému	34
Obrázok 10 Schéma tabuliek pre správu užívateľov	35
Obrázok 11 Schéma tabuliek pre regionálnu správu	35
Obrázok 12 Schéma tabuliek pre správu zjazdov	36
Obrázok 13 Schéma tabuliek pre správu objednávok.....	37
Obrázok 14 Schéma tabuliek pre správu zliav.....	38
 Tabuľka 1 Časový harmonogram	 30

9.3 Fyzická implementácia

Dátová štruktúra vrátane dátových typov jednotlivých atribútov.



9.4 Zdrojový kód na vytvorenie dátovej štruktúry

Štruktúra tabuliek v nasledujúcom kóde sa úplne nezhoduje s navrhovanou dátovou štruktúrou z dôvodu optimalizácie rýchlosti a pridaniu nových funkcií v čase po napísaní kapitoly Vlastné návrhy riešenia.

```
CREATE TABLE `art_article` (  
  `ID` int(11) NOT NULL AUTO_INCREMENT,  
  `AuthorID` int(11) NOT NULL DEFAULT '0',  
  `CategoryID` int(11) NOT NULL DEFAULT '0',  
  `CategoryPath` text CHARACTER SET latin1 NOT NULL,  
  `UID` varchar(50) NOT NULL DEFAULT '',  
  `Name` varchar(200) NOT NULL DEFAULT '',  
  `PreText` text NOT NULL,  
  `Text` text NOT NULL,  
  `Column` tinyint(4) NOT NULL DEFAULT '1',  
  `Order` int(11) NOT NULL DEFAULT '0',  
  `PageTitle` varchar(100) NOT NULL DEFAULT '',  
  `PageKeywords` text NOT NULL,  
  `Created` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',  
  `LastModified` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',  
  `Active` tinyint(1) NOT NULL DEFAULT '0',  
  `ActiveFrom` datetime DEFAULT NULL,  
  `ActiveTo` datetime DEFAULT NULL,  
  `Erased` tinyint(1) NOT NULL DEFAULT '0',  
  PRIMARY KEY (`ID`),  
  KEY `CategoryID` (`CategoryID`),  
  FULLTEXT KEY `Text` (`Text`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `cat_category` (  
  `ID` int(11) NOT NULL AUTO_INCREMENT,  
  `ParentID` int(11) DEFAULT '0',  
  `UID` varchar(50) NOT NULL DEFAULT '',  
  `Name` varchar(200) NOT NULL DEFAULT '',  
  `Path` varchar(200) CHARACTER SET latin1 NOT NULL DEFAULT '',  
  `PathName` text NOT NULL,  
  `Order` int(11) NOT NULL DEFAULT '0',  
  `Depth` tinyint(4) NOT NULL DEFAULT '0',  
  `ChildrenCount` int(11) NOT NULL DEFAULT '0',  
  `Type` tinyint(4) NOT NULL DEFAULT '1',  
  `PageType` tinyint(4) DEFAULT '0',  
  `Columns` tinyint(4) NOT NULL DEFAULT '1',  
  `Url` varchar(200) NOT NULL DEFAULT '',  
  `PageTitle` varchar(100) NOT NULL DEFAULT '',  
  `PageKeywords` text NOT NULL,  
  `Created` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',  
  `LastModified` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',  
  `Active` tinyint(1) NOT NULL DEFAULT '0',  
  `ActiveFrom` datetime DEFAULT '0000-00-00 00:00:00',  
  `ActiveTo` datetime DEFAULT '0000-00-00 00:00:00',  
  `Erased` tinyint(1) NOT NULL DEFAULT '0',  
  PRIMARY KEY (`ID`,`UID`),  
  KEY `ParentID` (`ParentID`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `cor_mail` (  
  `Id` int(11) NOT NULL DEFAULT '0',  
  `Uid` varchar(50) NOT NULL DEFAULT '',  
  `Name` varchar(100) NOT NULL DEFAULT '',  
  `FromAddress` varchar(100) NOT NULL DEFAULT '',  
  `FromDisplayName` varchar(100) NOT NULL DEFAULT '',
```

```

        `Subject` varchar(100) NOT NULL DEFAULT '',
        `XmlInputSample` text NOT NULL,
        `Template` text NOT NULL,
        `Active` tinyint(1) NOT NULL DEFAULT '0',
        PRIMARY KEY (`Id`)
    ) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `cor_setting` (
    `Key` varchar(100) NOT NULL DEFAULT '',
    `Description` varchar(200) NOT NULL DEFAULT '',
    `Value` varchar(100) NOT NULL DEFAULT '',
    `Editable` tinyint(1) NOT NULL DEFAULT '0',
    PRIMARY KEY (`Key`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

CREATE TABLE `reg_country` (
    `Id` int(11) NOT NULL AUTO_INCREMENT,
    `Name` varchar(50) NOT NULL DEFAULT '',
    `ArticleId` int(11) DEFAULT '0',
    `GoogleMapParameter` varchar(50) DEFAULT NULL,
    `DestinationCount` int(11) NOT NULL DEFAULT '0',
    `Created` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
    `LastModified` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
    `Active` tinyint(1) NOT NULL DEFAULT '0',
    PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `reg_destination` (
    `Id` int(11) NOT NULL AUTO_INCREMENT,
    `CountryId` int(11) NOT NULL DEFAULT '0',
    `Name` varchar(100) NOT NULL DEFAULT '',
    `ArticleId` int(11) DEFAULT '0',
    `GoogleMapParameter` varchar(50) DEFAULT NULL,
    `HotelCount` int(11) NOT NULL DEFAULT '0',
    `Created` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
    `LastModified` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
    `Active` tinyint(1) NOT NULL DEFAULT '0',
    PRIMARY KEY (`Id`),
    KEY `Country` (`CountryId`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `reg_hotel` (
    `Id` int(11) NOT NULL AUTO_INCREMENT,
    `CountryId` int(11) NOT NULL DEFAULT '0',
    `DestinationId` int(11) NOT NULL DEFAULT '0',
    `Name` varchar(100) NOT NULL DEFAULT '',
    `ArticleId` int(11) DEFAULT '0',
    `GoogleMapParameter` varchar(50) DEFAULT NULL,
    `ImageUrl` varchar(200) NOT NULL,
    `Created` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
    `LastModified` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
    `Active` tinyint(1) NOT NULL DEFAULT '0',
    PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_account` (
    `Id` int(11) NOT NULL AUTO_INCREMENT,
    `Name` varchar(200) NOT NULL DEFAULT '',
    `AccountNumber` varchar(20) NOT NULL DEFAULT '',
    `BankCode` varchar(10) NOT NULL DEFAULT '',
    PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_diettype` (
    `Id` int(11) NOT NULL AUTO_INCREMENT,
    `Name` varchar(50) NOT NULL DEFAULT '',

```



```

PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_discount` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `DiscountPackageId` int(11) NOT NULL DEFAULT '0',
  `TourProviderId` int(11) NOT NULL DEFAULT '0',
  `Name` varchar(100) NOT NULL DEFAULT '',
  `Type` tinyint(4) NOT NULL DEFAULT '0',
  `Percentage` decimal(10,0) DEFAULT NULL,
  `Value` decimal(10,0) DEFAULT NULL,
  `DateFrom` date DEFAULT NULL,
  `DateTo` date DEFAULT NULL,
  PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_discountpackage` (
  `Id` int(11) NOT NULL DEFAULT '0',
  `TourProviderId` int(11) NOT NULL DEFAULT '0',
  `Name` varchar(100) NOT NULL DEFAULT '',
  PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_hotelroom` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `HotelId` int(11) NOT NULL DEFAULT '0',
  `RoomTypeId` int(11) NOT NULL DEFAULT '0',
  `RegularBedCount` tinyint(4) NOT NULL DEFAULT '0',
  `ExtraBedCount` tinyint(4) NOT NULL DEFAULT '0',
  `RollawayCount` tinyint(4) NOT NULL DEFAULT '0',
  PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_order` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `UId` varchar(50) CHARACTER SET latin1 NOT NULL DEFAULT '',
  `Guid` varchar(50) NOT NULL DEFAULT '',
  `CountryId` int(11) NOT NULL DEFAULT '0',
  `DestinationId` int(11) NOT NULL DEFAULT '0',
  `HotelId` int(11) NOT NULL DEFAULT '0',
  `TourId` int(11) NOT NULL DEFAULT '0',
  `TermId` int(11) DEFAULT NULL,
  `DateFrom` datetime DEFAULT NULL,
  `DateTo` datetime DEFAULT NULL,
  `DiscountId` int(11) DEFAULT NULL,
  `State` tinyint(4) NOT NULL DEFAULT '0',
  `TotalPrice` decimal(10,2) DEFAULT NULL,
  `Commission` decimal(10,2) DEFAULT NULL,
  `FirstName` varchar(100) NOT NULL DEFAULT '',
  `LastName` varchar(100) NOT NULL DEFAULT '',
  `Address` varchar(200) NOT NULL DEFAULT '',
  `DateOfBirth` date NOT NULL DEFAULT '0000-00-00',
  `Email` varchar(50) NOT NULL,
  `PhoneNumber` varchar(20) NOT NULL,
  `Comment` text NOT NULL,
  `AdminComment` text NOT NULL,
  `Created` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_orderpassenger` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `OrderId` int(11) NOT NULL,
  `FirstName` varchar(100) NOT NULL,
  `LastName` varchar(100) NOT NULL,
  `DateOfBirth` date NOT NULL,
  `Address` varchar(200) NOT NULL DEFAULT '',

```

```

PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_orderservice` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `OrderId` int(11) NOT NULL,
  `TourServiceId` int(11) NOT NULL,
  `TourServicePrice` decimal(10,2) NOT NULL DEFAULT '0.00',
  `ApplyDiscount` tinyint(1) NOT NULL DEFAULT '0',
  `ClientAmount` int(11) NOT NULL DEFAULT '0',
  `TourProviderAmount` int(11) NOT NULL DEFAULT '0',
  PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_payment` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `OrderId` int(11) NOT NULL DEFAULT '0',
  `AccountId` int(11) DEFAULT NULL,
  `CounterAccountNumber` varchar(20) DEFAULT NULL,
  `CounterBankCode` varchar(10) DEFAULT NULL,
  `Receipt` varchar(20) NOT NULL DEFAULT '',
  `Amount` decimal(10,2) NOT NULL DEFAULT '0.00',
  `Paid` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  `NotificationSent` tinyint(1) NOT NULL DEFAULT '0',
  PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_price` (
  `TourId` int(11) NOT NULL DEFAULT '0',
  `TermId` int(11) NOT NULL DEFAULT '0',
  `TourServiceId` int(11) NOT NULL DEFAULT '0',
  `Value` decimal(10,2) DEFAULT NULL,
  `ValueAction` decimal(10,2) DEFAULT NULL,
  `SoldOut` tinyint(1) NOT NULL DEFAULT '0',
  PRIMARY KEY (`TourId`,`TermId`,`TourServiceId`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_roomtype` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `Name` varchar(50) NOT NULL DEFAULT '',
  PRIMARY KEY (`Id`)
) ENGINE=MyISAM AUTO_INCREMENT=5 DEFAULT CHARSET=utf8;

CREATE TABLE `trl_servicetype` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `Name` varchar(50) NOT NULL DEFAULT '',
  PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_term` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `HotelId` int(11) NOT NULL DEFAULT '0',
  `TourId` int(11) NOT NULL DEFAULT '0',
  `DateFrom` date NOT NULL DEFAULT '0000-00-00',
  `DateTo` date NOT NULL DEFAULT '0000-00-00',
  `LastMinute` tinyint(1) NOT NULL DEFAULT '0',
  `SoldOut` tinyint(1) NOT NULL DEFAULT '0',
  `Active` tinyint(1) NOT NULL DEFAULT '0',
  PRIMARY KEY (`Id`),
  KEY `Pricelist` (`TourId`),
  KEY `Active` (`Active`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_tour` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `TourProviderId` int(11) NOT NULL DEFAULT '0',
  `DiscountPackageId` int(11) DEFAULT NULL,

```

```

`CountryId` int(11) NOT NULL DEFAULT '0',
`DestinationId` int(11) NOT NULL DEFAULT '0',
`HotelId` int(11) NOT NULL DEFAULT '0',
`TourTypeId` int(11) NOT NULL DEFAULT '0',
`TransportTypeId` int(11) NOT NULL DEFAULT '0',
`DietTypeInPriceId` int(11) DEFAULT NULL,
`ArticleId` int(11) DEFAULT NULL,
`Name` varchar(100) NOT NULL DEFAULT '',
`Description` text NOT NULL,
`FixedTerms` tinyint(1) NOT NULL DEFAULT '0',
`PriceFrom` decimal(10,2) DEFAULT NULL,
`PriceFromDescription` varchar(100) NOT NULL DEFAULT '',
`Recommended` tinyint(1) NOT NULL DEFAULT '0',
`Created` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
`LastModified` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
`Active` tinyint(1) NOT NULL DEFAULT '0',
PRIMARY KEY (`Id`),
KEY `Hotel` (`HotelId`),
KEY `Destination` (`DestinationId`),
KEY `Country` (`CountryId`)
) ENGINE=MyISAM AUTO_INCREMENT=367 DEFAULT CHARSET=utf8;

CREATE TABLE `trl_tourdiettype` (
  `TourId` int(11) NOT NULL DEFAULT '0',
  `DietTypeId` int(11) NOT NULL DEFAULT '0',
  PRIMARY KEY (`TourId`,`DietTypeId`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_tourprovider` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `Name` varchar(100) NOT NULL DEFAULT '',
  PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_tourservice` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `TourId` int(11) NOT NULL DEFAULT '0',
  `ServiceTypeId` int(11) NOT NULL DEFAULT '0',
  `Name` varchar(50) NOT NULL DEFAULT '',
  `Description` varchar(100) NOT NULL DEFAULT '',
  `Order` tinyint(4) NOT NULL DEFAULT '0',
  `DisplayInPricelist` tinyint(1) NOT NULL DEFAULT '0',
  `ApplyDiscount` tinyint(1) DEFAULT '0',
  `Active` tinyint(1) NOT NULL DEFAULT '0',
  PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_tourtype` (
  `Id` int(11) NOT NULL DEFAULT '0',
  `Name` varchar(50) NOT NULL DEFAULT '',
  PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `trl_transporttype` (
  `Id` int(11) NOT NULL DEFAULT '0',
  `Name` varchar(50) NOT NULL DEFAULT '',
  PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `usr_right` (
  `Key` varchar(20) NOT NULL DEFAULT '',
  `Description` varchar(200) CHARACTER SET utf8 NOT NULL DEFAULT '',
  PRIMARY KEY (`Key`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `usr_subscriber` (
  `Email` varchar(100) NOT NULL DEFAULT '',

```

```

`UserId` int(11) DEFAULT NULL,
`Created` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
`ActivationCode` varchar(10) NOT NULL DEFAULT '',
`Activated` datetime DEFAULT NULL,
`Active` tinyint(1) NOT NULL DEFAULT '0',
`DeactivationCode` varchar(10) NOT NULL DEFAULT '',
`Deactivated` datetime DEFAULT NULL,
PRIMARY KEY (`Email`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `usr_user` (
`Id` int(11) NOT NULL AUTO_INCREMENT,
`Login` varchar(50) NOT NULL DEFAULT '',
`Salt` int(11) NOT NULL DEFAULT '0',
`Password` varchar(50) NOT NULL DEFAULT '',
`FirstName` varchar(100) NOT NULL DEFAULT '',
`LastName` varchar(100) NOT NULL DEFAULT '',
`Nickname` varchar(100) NOT NULL DEFAULT '',
`Email` varchar(100) NOT NULL DEFAULT '',
`Comment` text NOT NULL,
`SystemUser` tinyint(1) NOT NULL DEFAULT '0',
`AvatarUrl` varchar(100) NOT NULL DEFAULT '',
`CountryId` int(11) NOT NULL DEFAULT '0',
`Created` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
`LastLogin` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
`ActivationCode` varchar(10) NOT NULL DEFAULT '',
`Active` tinyint(1) NOT NULL DEFAULT '0',
PRIMARY KEY (`Id`),
UNIQUE KEY `Login` (`Login`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE `usr_userright` (
`UserId` int(11) NOT NULL DEFAULT '0',
`RightId` varchar(20) NOT NULL DEFAULT '',
KEY `UserId` (`UserId`),
KEY `RightId` (`RightId`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

```